



UNIVERSIDAD DE BUENOS AIRES  
FACULTAD DE CIENCIAS EXACTAS Y NATURALES  
DEPARTAMENTO DE COMPUTACIÓN

# Parcelando la Corteza Cerebral a través de Resonancia Magnética de Difusión

Tesis presentada para optar al título de  
Licenciado en Ciencias de la Computación

Guillermo Alejandro Gallardo Diez

Director: Demian Wassermann

Codirector:

Buenos Aires, 2011



## Índice general

1.. Introducción . . . . .	1
2.. Introducción a la resonancia magnética de difusión . . . . .	5
2.1. Imagen por resonancia magnética . . . . .	5
2.2. Resonancia magnética de difusión . . . . .	6
3.. Parcelando la corteza cerebral . . . . .	11
3.1. Adquiriendo datos de los sujetos . . . . .	11
3.2. Seleccionando voxels que serán semillas . . . . .	11
3.3. Creando tractogramas a partir de las semillas . . . . .	13
3.4. Clustering de semillas: Moreno-Dominguez . . . . .	14
4.. Contribuciones Metodológicas . . . . .	17
4.1. Estudio de la convergencia de los tractogramas . . . . .	17
4.2. Análisis del método Moreno-Dominguez . . . . .	19
4.2.1. Clustering de vectores colineales . . . . .	19
4.2.2. Relación métrica- <i>linkage</i> . . . . .	21
4.2.3. Complejidad algorítmica del clustering . . . . .	21
4.3. Transformando los tractogramas al espacio <i>LogOdds</i> . . . . .	22
4.3.1. Transformación Logit . . . . .	22
4.3.2. Modificando el algoritmo de clustering . . . . .	23
4.3.3. Almacenando tractogramas: Matrices ralas . . . . .	24
4.4. Parcelando la corteza en el espacio <i>LogOdds</i> . . . . .	26
4.4.1. Pre-clustering . . . . .	26
4.4.2. Algoritmo de clustering . . . . .	27
5.. Resultados . . . . .	29
5.1. Convergencia de los tractogramas . . . . .	29
5.2. Parcelamiento del área de Broca . . . . .	29
5.2.1. Método Moreno-Dominguez . . . . .	30
5.2.2. Parcelamiento en el espacio <i>LogOdds</i> . . . . .	32
5.2.3. Resultados de los métodos en mayor detalle . . . . .	33
5.3. Parcelamiento del hemisferio izquierdo . . . . .	33
5.3.1. Método Moreno-Dominguez . . . . .	34
5.3.2. Utilizando nuestro método . . . . .	34
5.3.3. Resultados de los métodos en mayor detalle . . . . .	35
6.. Discusión . . . . .	39
6.1. Convergencia del algoritmo de tractografía . . . . .	39
6.2. Comparación entre ambos métodos de <i>clustering</i> . . . . .	39
6.3. Correlato con otras parcelaciones de la literatura . . . . .	40
6.3.1. Correlación anatómica . . . . .	40
6.3.2. Correlación funcional . . . . .	42

6.4. Trabajo Futuro . . . . .	45
-------------------------------	----

## 1. INTRODUCCIÓN

En la neurociencia moderna existe la teoría de que el cerebro puede ser dividido en áreas de acuerdo a distintos criterios estructurales, siendo la citoarquitectura de Broadmann [1] una de las más conocidas. Actualmente existe evidencia de que es posible atribuir un criterio funcional a cada una de estas regiones, como mover la mano o procesar el lenguaje [2]. No obstante, la relación de dichas áreas con funciones complejas sigue siendo un problema abierto [3]. Así mismo, tampoco se conoce si esta división es única, y en tal caso, qué límites definen a cada área. La relación función-estructura posee diversas aplicaciones en la medicina, como por ejemplo el planeamiento quirúrgico [4] [5]; asistencia durante la cirugía [6] y rehabilitación [7] de pacientes. Por ello existe la necesidad de entender como es que el cerebro está conectado estructuralmente, esto es, entender qué regiones están conectadas físicamente por axones y cuánto influye esto en el aspecto funcional del cerebro.

El cerebro puede dividirse en varios tipos de tejido, nosotros nos enfocaremos en dos de ellos: materia blanca y materia gris. La corteza está formada por materia gris, la cual es densa en neuronas. Estas neuronas están conectadas entre sí mediante axones. Se denomina materia blanca al tejido donde la proporción de axones es superior a la de neuronas [8]. La reciente invención de la Resonancia Magnética de Difusión (dMRI) permitió desarrollar nuevas técnicas para estudiar la conectividad estructural [9]. En particular, el conocer la intensidad de difusión que existe en cada punto del cerebro permite caracterizar los axones dentro de la materia blanca [10]. Una forma de hacerlo es utilizando algoritmos de tractografía [11, 12], estos toman un punto del cerebro como semilla y devuelven un tractograma. Un tractograma es una imagen donde cada voxel representa la probabilidad de que ese punto del cerebro esté conectado a la semilla elegida mediante un conjunto de axones. Las probabilidades se estiman mediante un procedimiento Monte Carlo, simulando el recorrido de un número de partículas de agua por la materia blanca comenzando desde dicha semilla. Distintos grupos han propuesto el agrupar semillas para definir nuevos criterios de parcelamiento de la corteza cerebral. Esto se puede hacer empleando algoritmos de *clustering* sobre los tractogramas de las semillas.

Los algoritmos de *clustering* son herramientas de análisis en campos como *Machine Learning* y *Data Mining*. Permiten clasificar objetos en base a distintos criterios. Ejemplos de ellos son:

- K-means [13]: Divide  $n$  vectores en  $k$  *clusters* diferentes, siendo  $k$  un número definido *a priori*. Cada *cluster* está formado por los elementos que más cerca están al vector medio del *cluster*.
- Agglomerative Hierarchical Clustering [14]: Cada observación comienza en un *cluster* distinto. Luego, iterativamente el algoritmo selecciona dos *clusters* siguiendo algún criterio de similitud, los agrupa en un nuevo *cluster* y crea un elemento representativo de este. El proceso finaliza cuando solo queda un *cluster*. La jerarquía resultante es expresada mediante un dendrograma.

- Gaussian Mixture [14]: Asume que todas las observaciones provienen de un número finito de distribuciones Gaussianas con parámetros desconocidos. Implementa la técnica *expectation-maximization* para determinar dichos parámetros iterativamente.

Cada algoritmo utiliza distintos modelos de *cluster*, por lo que el espacio de aplicación y el resultado varía significativamente de uno a otro.

Para poder parcelar la corteza cerebral mediante el *clustering* de tractogramas es necesario seguir una serie de pasos. El primer paso es seleccionar la posición de las semillas de cada tractograma. Como la materia blanca está compuesta principalmente por axones, es en ella donde se sitúan las semillas. Por ello, seleccionar los voxels que serán semilla requiere contar con alguna manera de discriminar entre materia blanca y materia gris. Si además queremos que las semillas estén a cierta distancia de la corteza debemos tener cuidado. La corteza del cerebro no es uniforme, está llena de surcos y circunvoluciones como puede verse en la figura 1.1. El segundo paso es generar los tractogramas. En la literatura actual se utilizan hasta 20000 semillas por hemisferio [15] y 100000 partículas por semilla [16] para generar los tractogramas. Realizar cada tractograma en paralelo reduce el tiempo total del procesamiento del cerebro. Al momento de almacenar los tractogramas es importante seleccionar la estructura correcta. Una matriz del tamaño de la imagen de dMRI es la implementación más sencilla pero, como explicaremos en la sección 4.3.3, desperdicia mucho espacio. En tercer lugar, para agrupar los tractogramas se debe seleccionar un modelo para los datos; un algoritmo de *clustering* y una medida de similitud. Finalmente, una vez obtenidos los *clusters*, es necesario mapear cada semilla con su respectivo voxel en la corteza cerebral. En resumen, primero hay que situar las semillas en la materia blanca; luego se deben generar los tractogramas de estas; hay que agrupar dichos tractogramas y finalmente trasladar el *clustering* a la corteza cerebral.

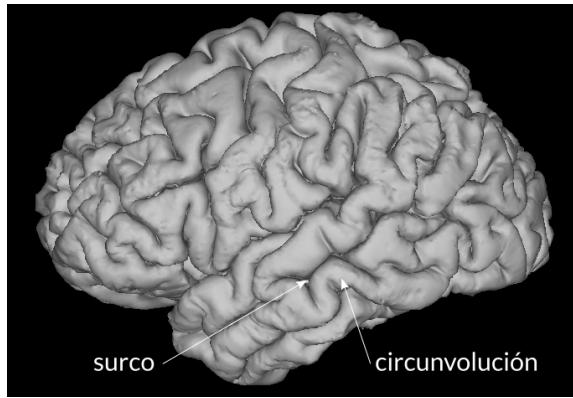


Fig. 1.1: La corteza cerebral posee pliegues denominados circunvoluciones que están separados por surcos.

Dada la cantidad de herramientas y configuraciones posibles, distintos grupos han aplicado diferentes criterios de *clustering* para parcelar la corteza. Por ejemplo, Behrens et. al [17] utilizan *Target-Based Clustering*. Este algoritmo define como restricción que cada parcela solo puede estar conectada con alguna otra perteneciente a un conjunto predefinido. Anwander et. al [16] dividen el área de Broca utilizando *k-means*, por lo cual necesitan predefinir el número de parcelas a encontrar. Moreno-Dominguez et al. [15] utilizan el algoritmo *Agglomerative Hierarchical Clustering* para parcelar la corteza cerebral. Eligen como medida de similitud la distancia coseno y como representante al centroide. La gran ventaja de este último método de *clustering* es que no restringe *a priori* el número de parcelas a generar o la forma en que estas deberían estar conectadas. A su vez, el elegir como cortar el dendrograma permite controlar la granularidad de la

---

parcelación. Sin embargo, no queda claro que el criterio utilizado por Moreno-Dominguez para medir distancias entre *clusters* y la forma de representar la unión de los mismos sean compatibles. Entonces, si bien el método de Moreno-Dominguez no asume cuantas o como son las áreas a encontrar, aún no posee el suficiente formalismo. Por esto es importante seguir investigando nuevos modelos.

Nuestro objetivo es analizar los métodos de *clustering* jerárquico actuales para parcelar toda la corteza y proponer un nuevo enfoque. Para ello implementamos dos métodos de *clustering* y los utilizamos sobre sujetos de la base de datos *Human Connectome Project* [18]. Ésta posee la dMRI de varios sujetos organizada por sexo y edad, con la ventaja de que todos los datos están ya preprocesados. Usar dicha base de datos nos permite enfocarnos en el problema del *clustering*. A su vez, permite reproducir con mayor facilidad nuestro estudio.

Una contribución menor de este trabajo es analizar la implementación existente del algoritmo de tractografía que usamos. Dada la naturaleza estocástica del mismo es necesario comprobar si el resultado se estabiliza al utilizar un número suficientemente grande de semillas. En particular mostramos que para mas de 2000 partículas los tractogramas convergen a un mismo resultado.

La contribución principal de este estudio es el realizar un análisis teórico del método de Moreno-Dominguez [15]; mostrar las problemáticas que presenta y proponer una nueva forma de parcelar la corteza mediante el *clustering* de semillas. Nuestro método consiste en transformar los tractogramas a un espacio vectorial y luego agruparlos con *Agglomerative Hierarchical Clustering*. Utilizamos la métrica euclidiana como medida de similitud y el centroide como *linkage*. Presentamos una implementación eficiente de nuestro método así como también el como adaptarlo al uso de matrices ralas para optimizar el costo espacial. Mostramos que nuestro algoritmo produce parcelaciones similares a las obtenidas usando el estado del arte. La ventaja de nuestro algoritmo es que reduce significativamente el orden de complejidad temporal y espacial. También mostramos que nuestros resultados poseen consistencia con parcelaciones anatómicas [39] y funcionales [3, 40] de la literatura.

Este trabajo está dividido en 6 capítulos. El capítulo 2 es una breve introducción teórica a la Resonancia Magnética y a la Resonancia Magnética de Difusión. El capítulo 3 detalla los pasos necesarios a seguir para parcelar la corteza cerebral y presenta el método de Moreno-Dominguez. El capítulo 4 muestra nuestras contribuciones: Explicamos el algoritmo de tractografía implementado y como estudiar su estabilidad; hacemos un análisis teórico del método de Moreno-Dominguez mostrando algunas de sus limitaciones; introducimos una transformación de los tractogramas que nos permite operar con ellos linealmente y presentamos nuestro método de parcelación en detalle. El capítulo 5 muestra los resultados de parcelar el área de Broca y la corteza cerebral con ambos métodos. Finalmente el capítulo 6 presenta la discusión de los resultados obtenidos junto con ideas para futuros trabajos.



## 2. INTRODUCCIÓN A LA RESONANCIA MAGNÉTICA DE DIFUSIÓN

Este capítulo está basado en el libro *Diffusion MRI* [19] y en las clases del Doctor Michael L. Lipton [20] disponibles online. En caso de querer profundizar en algún tema, por favor referirse a estos.

### 2.1. Imagen por resonancia magnética

Se denomina momento magnético nuclear al momento magnético que posee un átomo a causa del *spin* de sus protones y electrones. Cuando un protón con momento magnético  $\vec{\mu}$  es puesto dentro de un campo magnético comenzara a prececer en torno a la dirección de este último con una frecuencia:

$$\omega = \vec{\mu} \times \vec{B} = \gamma \vec{J} \times \vec{B}$$

Esta es la frecuencia de Larmor, donde  $\omega$  es la velocidad angular;  $\gamma$  es la relación giromagnética del protón;  $\vec{J}$  es su momento angular y  $B$  es la fuerza del campo. A su vez, la cantidad de energía del campo determinara el ángulo entre el momento magnético  $\vec{\mu}$  y el campo  $\vec{B}$  mientras sucede la precesión (figura 2.1). Esto quiere decir que dado un  $B$  suficientemente grande es posible hacer que la precesión suceda en la dirección transversal del campo, lo cual permitiría medir  $|\mu|$  simplemente poniendo una bobina en ese plano (figura 2.2). Si uno realizará el experimento y midiera  $|\mu|$  utilizando la bobina notaría que al apagar el campo, la señal comienza a desvanecerse. Esto es porque el sistema comienza a perder energía provocando que el ángulo entre  $\mu$  y el campo se achique. A este proceso se lo denomina relajación.

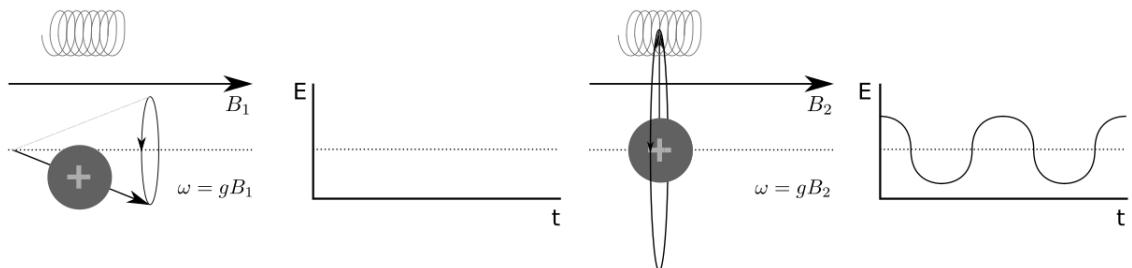


Fig. 2.1: Spin sometido a un campo magnético débil. La bobina no detecta el protón.

Fig. 2.2: Resultado de aumentar el campo magnético. La bobina detecta el protón.

Vayamos ahora al plano médico. El cuerpo está compuesto por distintos tipos de tejidos, cada uno con su propia composición química. Esto determina un momento magnético distinto para cada uno de ellos y por ende, un tiempo de relajación particular. Supongamos se pone una persona dentro de un campo magnético. Cada uno de sus tejidos comenzara a generar un momento en base a la población de protones que posea. Una forma de medir el tiempo de relajación de cada tejido sería trasladando la precesión al plano transversal. Si uno administrara la energía necesaria para esto simplemente aumentando la fuerza

del campo magnético dañaría al paciente. Aquí es donde se aprovecha fuertemente la frecuencia de Larmor. Conociendo la composición química de cada tejido es posible calcular de antemano su frecuencia angular. Luego, mediante el efecto de resonancia es posible transmitir energía al sistema simplemente emitiendo ondas en esa misma frecuencia.

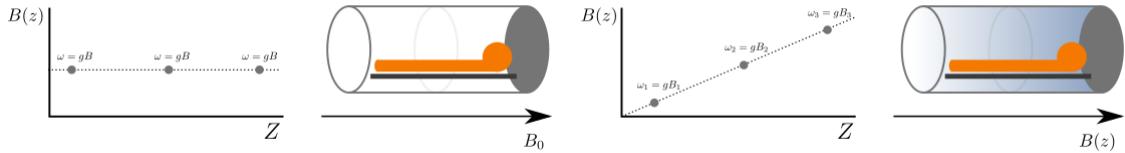


Fig. 2.3: Campo uniforme, todos los protones poseen la misma velocidad angular.

Fig. 2.4: Campo gradiente, la velocidad angular de los protones varía linealmente.

Los resonadores magnéticos son dispositivos con la capacidad de generar campos y pulsos en diferentes frecuencias. En particular todo resonador encendido está emitiendo constantemente un campo homogéneo  $B_0$  (figura 2.3). El problema entonces es: ¿Cómo obtener el tiempo de relajación de un punto particular del cuerpo?. Una respuesta posible es utilizar campos gradientes. Un campo gradiente es un campo que varía su potencia linealmente a lo largo de una dirección, provocando que todas las protones a lo largo de dicha dirección varíen su frecuencia angular de manera predecible. Aplicando un campo gradiente  $G_z$  en la dirección  $z$  (figura 2.4) sobre la persona haremos que la velocidad en función de la posición sea:  $\omega(z) = B_z(z)g$ , esto nos asegurara que si aplicamos un pulso de radio frecuencia (RF) con una frecuencia de  $B_z(z_o)g$ , solo los protones que se encuentran en la posición  $z = z_o$  comenzaran a resonar, por lo que estos serán los únicos que generen un campo transversal. Cabe destacar que como no es posible generar un pulso con exactamente la frecuencia deseada, también resonaran los protones que se encuentren cerca, por lo que tendremos un intervalo  $[z_o - \epsilon, z_o + \epsilon]$  resonando. A este proceso se lo denomina *slice selection*. Podemos pensar el *slice* como una matriz de dos dimensiones sobre el eje  $z$ . Si ahora aplicamos un campo gradiente  $G_\psi$  en la dirección  $y$ , sucederá que todas las filas de nuestra matriz adquirirán diferentes velocidades angulares. Al apagar  $G_\psi$  todos los protones volveran a preceder respecto al campo  $B_0$ , pero esta vez estarán desfasados por filas (figura 2.5). Encendiendo un tercer campo gradiente  $G_v$  sobre la dirección  $x$  conseguiremos que cada columna posea una frecuencia distinta y cada fila una fase distinta. Repitiendo este procedimiento varias veces cambiando solo la intensidad de  $G_\psi$  podemos armar lo que se conoce como *k-space*. El *k-space* es una imagen espacial temporal donde están anotados los valores obtenidos para cada potencia utilizada, en orden ascendente de potencia. El aplicar una transformada de Fourier 3D a dicho espacio nos devolverá la imagen que representa el contraste de cada tejido en el *slice* seleccionado.

## 2.2. Resonancia magnética de difusión

Las moléculas dentro de un fluido en equilibrio no se encuentran estáticas, sino que se mueven de manera aleatoria. A este fenómeno se lo conoce con el nombre de difusión.

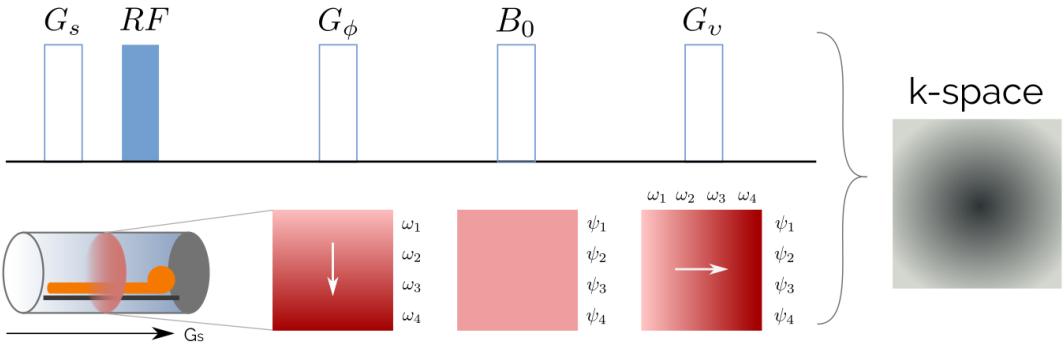


Fig. 2.5: Resumen del proceso de adquisición de imágenes en MRI.

En 1946 Bloch [21] prueba que la variación de la magnetización nuclear en el tiempo se puede expresar como:

$$\frac{dM(t)}{dt} = \gamma M(t) \times B(t) - \frac{M(t)\vec{x} + M(t)\vec{y}}{T_2} - \frac{(M(t) - M(0))\vec{z}}{T_1}$$

Donde  $\gamma$  es la relación giromagnética,  $B$  es la intensidad del campo magnético y  $T_1$ ,  $T_2$  son tiempos de relajación. Mas tarde, en 1956, H.C. Torrey [22] observa que la magnetización también se pierde por efecto de la difusión y extiende la ecuación de Bloch:

$$\frac{dM(t)}{dt} = gM(t) \times B(t) - \frac{M(t)\vec{x} + M(t)\vec{y}}{T_2} - \frac{(M(t) - M(0))\vec{z}}{T_1} + \nabla \cdot D \nabla M(t)$$

Esta relación se conoce como la ecuación de Bloch-Torrey.  $D$  es el tensor de difusión.

Imaginemos el siguiente experimento: luego de aplicar el pulso RF agregamos un campo gradiente  $G_1 = G_d$  durante un tiempo  $\delta$  pequeño. Como ya explicamos, esto generará un desfase entre los *spines* de los protones. El aplicar  $G_2 = -G_d$  luego de  $\Delta$  debería provocar que los *spines* se vuelvan a alinear. Sin embargo, los protones que se encuentren en un fluido estarán sometidos al efecto de la difusión. Esto sucede, por ejemplo, en el interior de los axones. Dependiendo del tiempo  $\delta$  los protones se habrán movido cierta distancia, provocando que el campo magnético los alcance en distintas posiciones. Por ende su velocidad angular se verá afectada de manera distinta a la esperada si no se hubieran movido. Esto nos indica que si hay difusión entonces habrá un desfase en esa población de neutrones (figura 2.6).

La señal medida en un resonador magnético proviene del momento magnético de los protones. Es importante destacar que por limitaciones físicas de los dispositivos es imposible obtener la señal producida por un solo protón. Lo que se mide es la resultante de los momentos magnéticos de todos los protones dentro de un espacio. Si todos los protones están precediendo a la misma velocidad sobre el mismo plano, entonces la resultante máxima se obtiene cuando todos poseen la misma fase. Esto es, todos se encuentran en la misma posición al mismo tiempo, rotando juntos. Por ende, el desfase producto de la

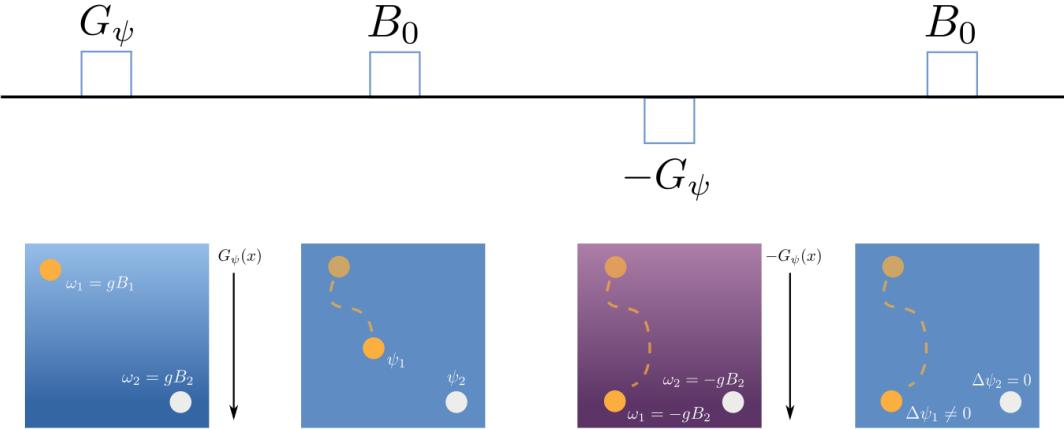


Fig. 2.6: Modificar la secuencia de gradientes permite medir la intensidad de difusión.

difusión se traducirá en perdida de señal.

En 1965 Stejskal y Tanner [23] crean una secuencia denominada *Pulsed Gradient Spin Echo*. En la misma utilizan dos pulsos de RF y un gradiente magnético para generar un desfase entre los protones (figura 2.7). Luego demuestran que asumiendo un  $\delta$  pequeño la ecuación de Bloch-Torrey tiene solución, y la atenuación de la señal se puede expresar como:

$$E(g, \delta, \Delta) = \frac{S(g, \delta, \Delta)}{S_0} = e^{-\gamma^2 g^2 \delta^2 (\Delta - \frac{\delta}{3}) D} \quad (2.1)$$

Donde  $E(g, \delta, \Delta)$  es la atenuación de la señal obtenida;  $g$  es la intensidad del gradiente magnético;  $S_0$  es la señal obtenida sin utilizar gradientes ( $g = 0T/m$ );  $\gamma$  es la relación giromagnética del protón;  $\delta$  es el tiempo que el gradiente está encendido;  $\Delta$  es el tiempo entre las activaciones del gradiente y  $D$  representa el coeficiente de difusión. La razón por la cual se divide la señal obtenida por  $S_0$  es porque la señal en cada punto depende fuertemente de la densidad de protones que hay en el mismo, si no ponderamos dicha densidad, es imposible comparar la intensidad de difusión en distintas regiones.

En 1985 Le Bihan [24] adapta esta técnica para medir la difusión de las partículas agrupando todos los parámetros del experimento dentro de un mismo parámetro  $b$ :

$$b = \gamma^2 g^2 \delta^2 \left( \Delta - \frac{\delta}{3} \right)$$

Esto simplifica la ecuación 2.1 a:

$$E(b) = \frac{S(b)}{S_0} = e^{-bD}$$

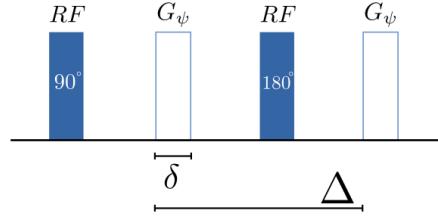


Fig. 2.7: Secuencia Pulsed Gradient Spin Echo.

Donde  $b$  representa el reciproco de la intensidad de difusión.

En 1994 Basser et al. [25] proponen medir la atenuación de señal en distintas direcciones y luego aproximar el coeficiente de difusión con un tensor de segundo orden. Un tensor es una matriz multidimensional asociado a una base, que posee una ley de transformación para indicar cómo cambian los componentes del tensor al cambiar de base. Esta técnica sienta las bases de lo que se conoce como *Diffusion Tensor Imaging* (DTI). En DTI el tensor más utilizado representa un elipsoide en  $R^3$ . La matriz que lo representa es simétrica, por ello es que se necesitan tomar al menos seis adquisiciones:

$$D = \begin{pmatrix} D_{xx} & D_{xy} & D_{xz} \\ D_{xy} & D_{yy} & D_{yz} \\ D_{xz} & D_{yz} & D_{zz} \end{pmatrix}$$

Uno de los principales limitantes de este método es que no permite representar correctamente el cruce de fibras. Esto es producto de que caracteriza las fibras dentro de cada voxel utilizando un único elipsoide.

En 1991 Callaghan et al [26] desarrollan el *q-space analysis*. Esto permite realizar microscopía mediante dMRI. Utilizando el trabajo de Stejskal y Tanner prueban que es posible obtener la siguiente relación entre la señal atenuada y una transformada de Fourier:

$$E(q, \Delta) = \frac{S(q, \Delta)}{S_0} = \int_{R^2} p(r; \Delta) e^{-2\pi i qr} dr$$

$$q = \frac{\gamma \delta g}{2\pi}$$

Donde  $p(r; t)$  es la densidad de probabilidad de que una población de partículas se desplace en dirección  $r$  durante un tiempo  $t$ .  $p(r; t)$  es característico del comportamiento donde se mueven las partículas.

Una de las principales ventajas de *q-space* sobre DTI es que no asume ningún modelo a priori, esto permite definir distintos tipos de modelos para  $p(r, t)$  que caracterizan mejor el cruce de fibras. *Spherical Harmonics* [27] o *Constrained Spherical Deconvolution* [28]. son ejemplos de ello.



### 3. PARCELANDO LA CORTEZA CEREBRAL

En el capítulo anterior presentamos los fundamentos físicos de la Resonancia Magnética Nuclear y como es posible utilizar estos para caracterizar la difusión en el cerebro. Nuestro objetivo ahora es parcelar la corteza cerebral haciendo uso de un criterio estructural. En particular, utilizando una imagen de difusión, queremos generar una parcelación mediante el agrupamiento de tractogramas. Para ello es necesario primero seleccionar los voxels que serán utilizados como semilla de cada tractograma; luego generar cada tractograma y finalmente agruparlos usando algún algoritmo de clustering. Dejamos el siguiente diagrama para ser utilizado como referencia de los pasos a seguir:



Fig. 3.1: Pipeline general del proceso de parcelación de la corteza cerebral.

#### 3.1. Adquiriendo datos de los sujetos

El primer paso para parcelar la corteza cerebral de un sujeto es conseguir tanto su imagen anatómica como de difusión. Actualmente existen bases de datos de gran calidad como es por ejemplo *Human Connectome Project*. [18] Las ventajas de utilizar estos datos son muchas: Tanto la imagen de difusión como la anatómica se encuentran ya preprocesadas [29]; cada sujeto posee una parcelación que, entre otras cosas, permite separar la materia blanca de la gris; y cada sujeto posee una superficie que representa su corteza cerebral. A su vez, hacer uso de una base de datos pública permite reproducir con mayor facilidad el estudio realizado.

#### 3.2. Seleccionando voxels que serán semillas

El segundo paso para parcelar la corteza cerebral es seleccionar que voxels serán semillas. La materia gris está compuesta principalmente por neuronas, y la materia blanca por axones que las comunican [8]. Como cada neurona posee asociado un axón, colocar semillas en la interfaz entre la materia gris y la blanca permite caracterizar las neuronas de la corteza [30] [16]. Cibu et al. [31] muestran que la materia blanca cercana a la materia gris está interconectada por pequeños axones. Como estamos interesados en realizar un estudio de las conexiones entre regiones distantes del cerebro, decidimos situar las semillas a  $3\text{mm}$  de la corteza, evitando así el efecto de éstos axones locales. El problema es que la corteza

del cerebro no es uniforme, sino que está llena de surcos y circunvoluciones. Calcular la distancia entonces no es inmediato, necesita un método que tome estas propiedades en cuenta. A continuación presentamos el método *Fast Marching Method* y como es posible utilizarlo para posicionar las semillas respetando la forma de la materia blanca.

*Fast Marching Method* es un método para resolver numéricamente una versión restringida de la ecuación *Eikonal*. La misma, en su forma general, es una ecuación diferencial no lineal que se encuentra comúnmente en problemas de propagación de onda. Tiene la forma:

$$V(x)|\nabla u(x)| = F(x), x \in \Omega$$

Donde  $\Omega$  es un subconjunto abierto de  $R^n$  con un *buen comportamiento* en su borde.  $F(x)$  se denomina el costo temporal y  $V(x)$  es la velocidad de la onda en cada punto. En el caso particular que queremos resolver  $u(x_\omega) = 0, x \in \delta\Omega$ ;  $F(x) = 1$  y  $V(x) = 1$ , por lo que la ecuación se resume a:

$$|\nabla u(x)| = 1, x \in \Omega$$

$u(v)$  en este caso representa el tiempo que tarda la onda en llegar desde algún elemento del borde hasta el punto  $v$  moviéndose a velocidad constante de una unidad de espacio por unidad de tiempo. Dada la forma de la velocidad,  $u(v)$  también representa **la distancia mas corta que existe entre cualquier punto  $v$  de la imagen y el borde de  $\Omega$** . Dependiendo la orientación que se elija, las distancias a los puntos internos de la superficie serán negativas y las distancias a los puntos externos positivas (figura 3.2). *FMM* encuentra estas distancias en tiempo  $O(n \log(n))$  [32], siendo  $n$  la cantidad de voxels de la imagen.

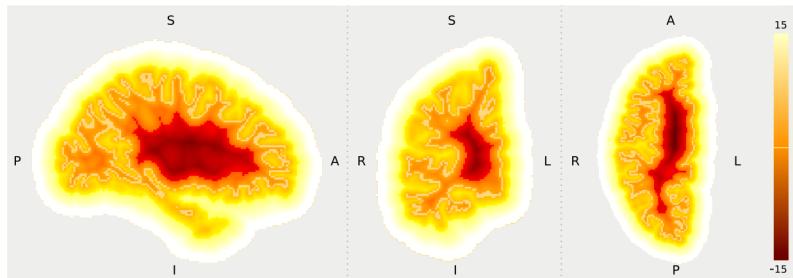


Fig. 3.2: *FMM* sobre el hemisferio derecho. El borde la materia blanca fue resaltado intencionalmente. Las distancias a los puntos internos son negativas y las distancias a los puntos externos positivas.

Es posible utilizar este algoritmo para seleccionar voxels a cierta profundidad en la materia blanca. Usando como borde la corteza cerebral podemos crear un mapa de distancias en la materia blanca. El gradiente de este mapa de distancias es un campo vectorial donde cada vector apunta hacia el interior de la materia blanca. Caminar partiendo desde los puntos en la siguiendo este campo permite adentrarse respetando la morfología de la materia blanca. Una ventaja de este método es que permite guardar un mapeo entre cada coordenada de la superficie y la semilla que la representa. Otra ventaja es que es posible realizar todo el proceso en tiempo  $O(n \log(n))$ .

### 3.3. Creando tractogramas a partir de las semillas

En la sección anterior mostramos como seleccionar voxels dentro de la materia blanca. El siguiente paso para parcelar la corteza cerebral es generar los tractogramas de estas semillas. Un tractograma es una imagen donde cada voxel representa la probabilidad de que ese punto del cerebro esté conectado a la semilla elegida mediante un conjunto de axones. Una forma de crear el tractograma de una semilla es computar un mapa de transiciones probabilístico entre los voxels del cerebro; luego generar un gran número de *streamlines* desde la semilla y finalmente calcular la fracción de partículas que pasaron por cada voxel. Se denomina *streamline* al camino que puede realizar una partícula de agua siguiendo un mapa probabilístico de transiciones entre voxels. El siguiente pseudocódigo denota este proceso:

---

#### Algoritmo 1 Proceso de creación de un tractograma

---

```

1: procedure CREAR_TRACTOGRAMA(ANATOMICA: IMG. ANATÓMICA,
2:                               DMRI: IMG. DE DIFUSIÓN,
3:                               S: SEMILLA, P: NUM. PARTÍCULAS,
4:                               STEP: TAM. DE PASO)
5:   mapa_probabilistico ← generar mapa desde DMRI
6:   mapa_visitas ← matriz vacía con las dimensiones de anatomica
7:   for p in [1...P] do
8:     streamlinep ← ∅
9:     posicion_actual ← posición de S
10:    while not criterio de parada do
11:      dirección_actual ← elegir dirección desde mapa_probabilistico
12:      posición_actual ← avanzar una distancia step en dirección_actual
13:      streamlinep ← agregar posición_actual
14:      for pos in streamlinep do
15:        mapa_visitas[pos] ← mapa_visitas[pos] + 1
16:   tractograma ← mapa_visitas / P
17:   return tractograma

```

---

Es importante destacar que el crear los tractogramas de esta manera genera un sesgo respecto a la distancia. Cuanto más lejos está un voxel mayor es el número de transiciones probabilísticas necesarias para llegar a él. Esto provoca que los voxels lejanos tengan valores pequeños. Para quitar este sesgo Moreno-Dominguez et al. [15] proponen crear los tractogramas de la siguiente manera:

$$T_i = \frac{\log(M_i + 1)}{\log(p + 1)} \quad (3.1)$$

Donde  $T_i$  es el valor del voxel  $i$  en el tractograma resultante;  $M_i$  es el valor del voxel  $i$  en el mapa de visitas y  $p$  es el número de partículas usadas. Dado que  $M_i \leq p, \forall i \in [1..N]$  cada voxel del tractograma tendrá un valor entre uno y cero.  $T_i = 1$  representa que todos los *streams* pasaron por el voxel, mientras que  $T_i = 0$  representa que ninguno pasó.

### 3.4. Clustering de semillas: Moreno-Dominguez.

En la sección anterior mostramos como crear tractogramas. El cuarto paso para parcelar la corteza cerebral es agrupar estos tractogramas. Moreno-Dominguez et al. [15] implementan el algoritmo *Agglomerative Hierarchical Clustering*. En este algoritmo, cada *feature* comienza en un *cluster* distinto. Luego, el algoritmo selecciona iterativamente dos *clusters* siguiendo algún criterio de similitud; los agrupa en un nuevo *cluster* y crea un elemento representativo de este. La jerarquía resultante de agrupar todos los *clusters* es expresada como un dendrograma. En el trabajo de Moreno-Dominguez utilizan como medida de similitud la distancia coseno (ecuación 3.2) y como criterio de *linkage* el centroide (ecuación 3.3).

$$\text{simil}_{\cos}(X, Y) = 1 - \frac{X \cdot Y}{\|X\| \|Y\|} \quad (3.2) \quad \text{centroide}(X, Y) = \frac{n_X X + n_Y Y}{n_X + n_Y} \quad (3.3)$$

$$X, Y \in R^m, n_z = \#z$$

Para crear los tractogramas transforman los mapas de visitas mediante la ecuación 3.1. Luego cambian a cero todos los voxels que poseen un valor menor a 0,4. Si se usan 15000 partículas, entonces estos voxels fueron visitados por menos del 0.3 % de ellas.

---

#### Algoritmo 2 Modificaciones al algoritmo Agglomerative Hierarchical Clustering.

---

```

1: procedure CLUSTERING(K_PASOS: PRIMEROS K PASOS,
2:                                TRACTOGRAMAS: MAT. DE TRACTOGRAMAS,
3:                                VECINOS: MAT. DE VECINOS,
4:                                DISTANCIAS: MAT. DE DISTANCIA CLUSTERS )
5:   jerarquia  $\leftarrow \emptyset$ 
6:   for k in [1, cant(tractogramas)] do
7:     if k > k_pasos then
8:       Cx, Cy  $\leftarrow$  clusters tales que x e y poseen distancia minima
9:     else
10:      Cx, Cy  $\leftarrow$  x e y son vecinos; de distancia minima y de tamaño similar.
11:      tractogramas  $\leftarrow$  eliminar clusters Cx, Cy
12:      centroide  $\leftarrow$  computar explicitamente el centroide (Cx, Cy)
13:      tractogramas  $\leftarrow$  agregar centroide
14:      jerarquia  $\leftarrow$  agregar la union (Cx, Cy)
15:      for Cz in tractogramas do
16:        D  $\leftarrow$  computar explicitamente distancia coseno de Cz al centroide
17:        distancias  $\leftarrow$  actualizar distancia entre (Cx, Cy) y centroide con D
18:   return jerarquia

```

---

Moreno-Dominguez et al. realizan varias modificaciones al algoritmo *Agglomerative Hierarchical Clustering* para mejorar su resultado. Aquí daremos solo una breve descripción de los mas relevantes, para mayores detalles favor de referirse al paper.

Una de las mayores modificaciones es agregar un parámetro para cambiar las primeras iteraciones del algoritmo. Dado un número *k*, las primeras *k* uniones son entre *clusters*

vecinos y de tamaño similar. Esto es, solo los *clusters* que se encuentran a menos de cierta distancia física en el cerebro pueden ser unidos. A su vez, solo se unen los *clusters* que poseen un tamaño similar para que el dendrograma crezca de manera balanceada. Estos cambios se pueden ver reflejados en el algoritmo 2.

Otra modificación al algoritmo de *clustering* les permite quitar *outliers* del dendrograma resultante. Evitan que los *clusters* de un solo elemento se unan a otros *clusters* si la (di)similitud es mayor a cierto *threshold*. Al hacer este paso durante el *clustering* previenen que los *outliers* afecten la forma de los nuevos centroides.

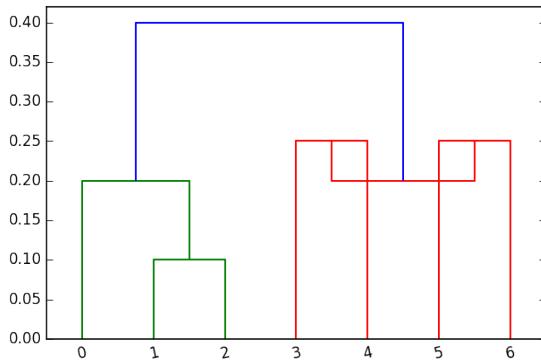


Fig. 3.3: Dendrograma con una inversión.

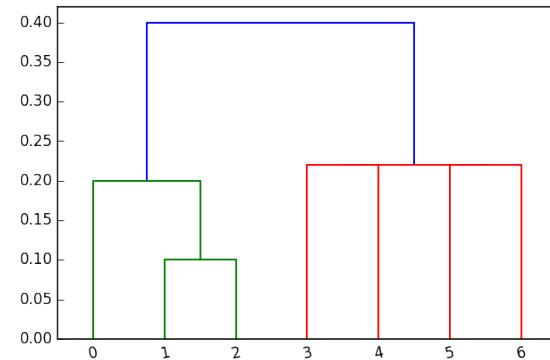


Fig. 3.4: Dendrograma con inversión corregida.

Una vez obtenido el dendrograma proceden a eliminar las inversiones dentro del mismo. Una inversión sucede cuando se unen dos *clusters* con una distancia interna mayor a la distancia entre ellos. Las inversiones no cambian la jerarquía de los *clusters*, sino que solo complican la interpretación visual de los datos [33]. Una forma de eliminarlas es colapsando las ramas que la componen en una sola jerarquía con mas de dos elementos. Un ejemplo de inversión y el resultado de quitarla se muestran en las Figuras 3.3 y 3.4 respectivamente.

Concluidos todos estos pasos se obtiene un dendrograma que representa el agrupamiento de los tractogramas de manera jerárquica. Para parcelar la corteza solo es necesario seleccionar una altura en la cual cortar dicho dendrograma. Los *clusters* que estén por debajo de ese corte formaran las distintas parcelas.



## 4. CONTRIBUCIONES METODOLÓGICAS

En el capítulo anterior presentamos como generar los tractogramas y el estado del arte para agruparlos. En este capítulo presentamos las contribuciones de nuestro trabajo. Comenzamos con una contribución menor que es estudiar la estabilidad del algoritmo de tractografía que utilizamos. Luego hacemos un análisis teórico del estado del arte y mostramos algunas de las problemáticas que posee. Finalmente utilizamos una función para transformar los tractogramas a un espacio vectorial y mostramos como esto permite mejorar tanto la complejidad espacial como temporal del algoritmo de *clustering*.

### 4.1. Estudio de la convergencia de los tractogramas

El diagrama de la figura 3.1 divide el proceso de parcelación de la corteza en varios pasos. El tercero de ellos consiste en generar tractogramas. Un tractograma para una semilla  $s$  es una imagen donde cada voxel  $v$  representa la probabilidad de que  $v$  esté conectado a  $s$  mediante un conjunto de axones. En la sección 3.3 presentamos una forma de crear tractogramas mediante un procedimiento Monte Carlo generando *streamlines*. Un *streamline* representa el movimiento aleatorio de una partícula de agua dentro de la materia blanca. Los tractogramas creados de esta manera son inherentemente estocásticos. Esto genera algunas preguntas interesantes: ¿Al repetir el experimento, podremos obtener el mismo tractograma? y ¿Cuántas partículas son necesarias para ello?. Para contestar estas preguntas utilizamos la técnica estadística de *bootstrap* [34].

---

#### Algoritmo 3 Algoritmo de tractografía utilizado

---

```
1: procedure TRACTOGRAMA(ANATOMICA: IMG. ANATÓMICA,  
2:                               DMRI: IMG. DE DIFUSIÓN,  
3:                               S: SEMILLA, P: NUM. PARTÍCULAS)  
4:      $csd \leftarrow$  Constrained Spherical Deconvolution Model of  $dmri$ .  
5:      $mapa\_probabilistico \leftarrow$  Mapa desde los coeficientes Spherical Harmonics en  $csd$   
6:      $mapa\_visitas \leftarrow$  matriz nula con mismas dimensiones que la anatómica  
7:     for  $p$  in  $[1\dots P]$  do  
8:          $streamline_p \leftarrow \emptyset$   
9:          $posicion\_actual \leftarrow$  posicion de  $S$   
10:        while  $posicion\_actual$  en materia blanca do  
11:             $direccion\_actual \leftarrow$  elegir direccion desde  $mapa\_probabilistico$   
12:             $posicion\_actual \leftarrow$  avanzar solo un voxel en  $direccion\_actual$   
13:             $streamline_p \leftarrow$  agregar  $posicion\_actual$   
14:        for  $pos$  in  $streamline_p$  do  
15:             $mapa\_visitas[pos] \leftarrow mapa\_visitas[pos] + 1$   
16:         $tractograma \leftarrow \log(mapa\_visitas+1) / \log(P+1)$   
17:    return  $tractograma$ 
```

---

*Bootstrap* permite estimar la distribución de un estadístico en base a calcularlo sobre remuestreos con repeticiones de una población. El método es especialmente útil cuando el número de muestras que se posee no es significativamente alto. Nosotros queremos estimar la distribución de la media de los *tractogramas* en función de la cantidad de *streamlines*. En la literatura se usan hasta 100000 *streamlines* por cada semilla [16]. *Bootstrap* nos permite realizar este estudio con muchas menos.

Para entender mejor la relación entre *streamlines* y tractogramas presentamos en el algoritmo 3 el pseudocódigo de la implementación de tractografía utilizada. Dicho algoritmo es una instancia del presentado en la sección 3.3. La imagen de difusión se enmarca en el modelo *Constrained Spherical Deconvolution* usando la forma propuesta por Aganj et al. [35]; el mapa de transiciones probabilísticas se recupera usando *Spherical Harmonics* [36] y el tractograma se calcula usando la ecuación 3.1 propuesta en el trabajo de Moreno-Dominguez et al. [15].

---

#### Algoritmo 4 Procedimiento de *Bootstrapping*

---

```

1: procedure ESTABILIDAD(S: SEMILLA)
2:   streamlines  $\leftarrow \emptyset$ 
3:   loop 15000 veces
4:     stream  $\leftarrow$  crear streamline desde S
5:     streamlines  $\leftarrow$  agregar stream
6:     medias  $\leftarrow \emptyset$ 
7:     varianzas  $\leftarrow \emptyset$ 
8:     for ss_size in [200, 500, 800, ...] do
9:       subsample  $\leftarrow \emptyset$ 
10:      loop 10000 veces
11:        tractograma  $\leftarrow$  tomar ss_size streams de streamlines y crear tractograma
12:        subsample  $\leftarrow$  agregar tractograma
13:        medias  $\leftarrow$  agregar tractograma medio de subsample
14:        varianzas  $\leftarrow$  agregar tractograma varianza de subsample
15:   return medias, varianzas

```

---

Usando el algoritmo 3 generamos 15000 *streamlines* para distintas semillas en el área de Broca. Luego calculamos el tractograma medio y la varianza de cada voxel utilizando 10000 submuestras aleatorias del mismo tamaño. Repetimos esto con varios tamaños de submuestra para estudiar la variabilidad del estadístico. Este procedimiento se puede ver en el algoritmo 4.

Crear los tractogramas es uno de los pasos necesarios para parcelar la corteza. Por la forma en que están definidos, los tractogramas son inherentemente estocásticos. El método aquí presentado nos permite estudiar la estabilidad del algoritmo de tractografía implementado. De esta manera podemos comprobar si dado un número suficiente de partículas los tractogramas de una semilla convergen a un mismo resultado.

## 4.2. Análisis del método Moreno-Dominguez

El cuarto paso del diagrama en la figura 3.1 es agrupar los tractogramas mediante un algoritmo de *clustering*. Moreno-Dominguez et al. [15] implementan el algoritmo *Agglomerative Hierarchical Clustering* para agrupar los tractogramas. En particular utilizan como medida de similitud la distancia coseno (ecuación 3.2) y como criterio de *linkage* el centroide (ecuación 3.3). Utilizar *Hierarchical Agglomerative Clustering* con la distancia coseno y el *linkage* centroide presenta algunas desventajas. Primero, la distancia coseno no distingue distancias entre clusters colineales. Segundo, el promedio de probabilidades no necesariamente representa una probabilidad [37]. Esto quiere decir que el centroide de un grupo de tractogramas no necesariamente representa el promedio de estos tractogramas. Finalmente la distancia coseno obliga a tener que comparar explícitamente cada centroide con los clusters existentes. Esto aumenta la complejidad temporal y espacial ya que es necesario mantener los tractogramas en memoria. A continuación detallamos cada una de estas desventajas.

### 4.2.1. Clustering de vectores colineales

La distancia coseno (ecuación 3.2) es una forma de medir correlación entre vectores. Supongamos tenemos vectores aleatorios colineales donde cada componente es independiente y proviene de una distribución de Bernulli. Agruparlos utilizando como medida de similitud la distancia coseno lleva a resultados erróneos. Por ejemplo, agrupar los puntos de la figura 4.1 usando el método de Moreno-Dominguez da como resultado la figura 4.2. Podemos ver que si bien teníamos tres clusters bien definidos en el espacio el método no logró separarlos.

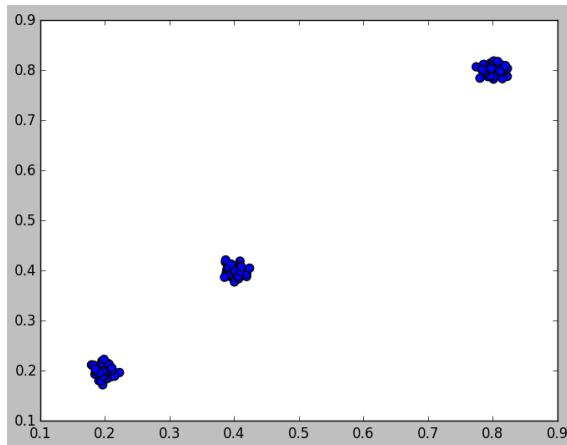
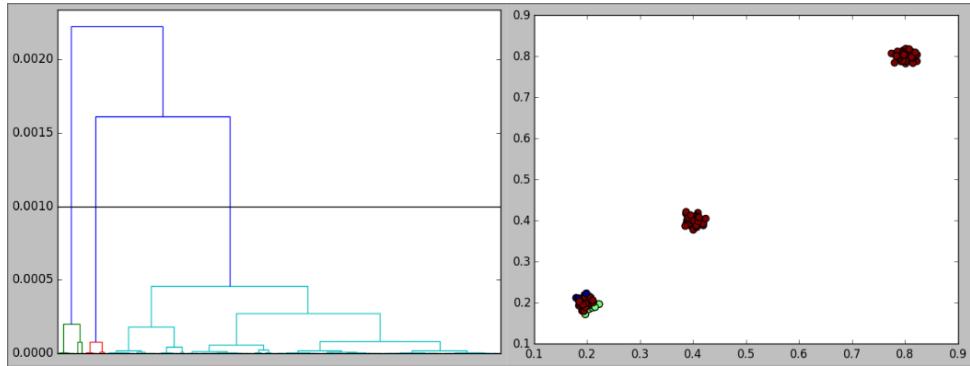


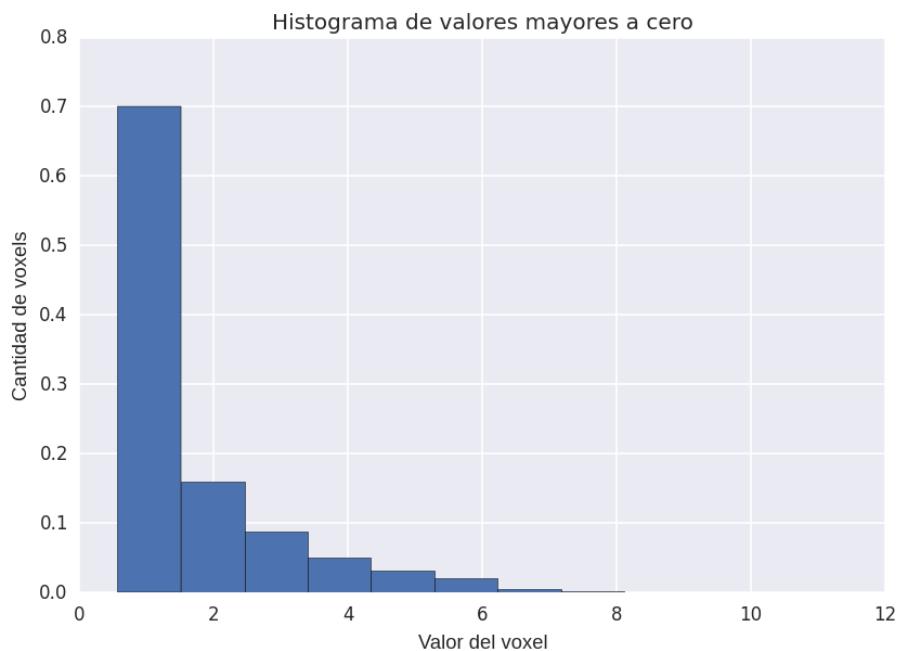
Fig. 4.1: Tres clusters colineales. Cada punto representa un vector aleatorio con componentes provenientes de una distribucion Bernulli. Queremos agruparlos utilizando Agglomerative Hierarchical Clustering.

También existe otro problema. Por la forma que posee el algoritmo 3, los tractogramas creados poseen muchos voxels lejanos con valores pequeños. El algoritmo simula el recorrido de partículas de agua sobre mapas de transiciones probabilísticas. Cuanto mas lejano un voxel mayor cantidad de transiciones se necesitan para llegar. Por lo tanto, cuanto mas lejos se encuentra un voxel, menor probabilidad tiene de ser visitado. Para visualizar esto

mejor situamos semillas en el área de Broca de un sujeto y generamos sus tractogramas. La figura 4.3 muestra el histograma de los valores en los mapas de visitas. Cada voxel dentro de un mapa de visitas indica que cantidad de partículas pasaron por él. En este caso el 70 % de los voxels visitados fueron alcanzados por solo una partícula. Esta gran cantidad de voxel con valores tan bajos podría generar ruido en las correlaciones.



*Fig. 4.2: Clustering resultante de utilizar el método de Moreno-Dominguez para agrupar los vectores. El algoritmo agrupa la mayoría de los vectores en una única categoría.*



*Fig. 4.3: Histograma normalizado de los valores en los mapas de visitas del área de Broca. No se muestran los valores mayores a 12. La mayor parte de los voxels visitados tuvieron una única visita.*

#### 4.2.2. Relación métrica-linkage

Moreno-Dominguez et al. utilizan como medida de similitud la distancia coseno (ecuación 3.2) y como criterio de *linkage* el centroide (ecuación 3.3). La medida de similitud compara el ángulo de los vectores, pero el *linkage* crea un representante minimizando la distancia euclídea a ambos clusters. No parecieran ser compatibles. Veamos un ejemplo. La figura 4.4 muestra 4 vectores aleatorios  $p_i = (X_i, Y_i)$ . Las variables  $X_i$  e  $Y_i$  son independientes y provienen de una distribución Bernulli. Las posiciones en coordenadas polares de cada vector son:  $p_1 = (0,4, 45^\circ)$ ;  $p_2 = (0,3, 25^\circ)$ ;  $p_3 = (0,4, 66^\circ)$  y  $p_4 = (0,4, 4,5^\circ)$ .

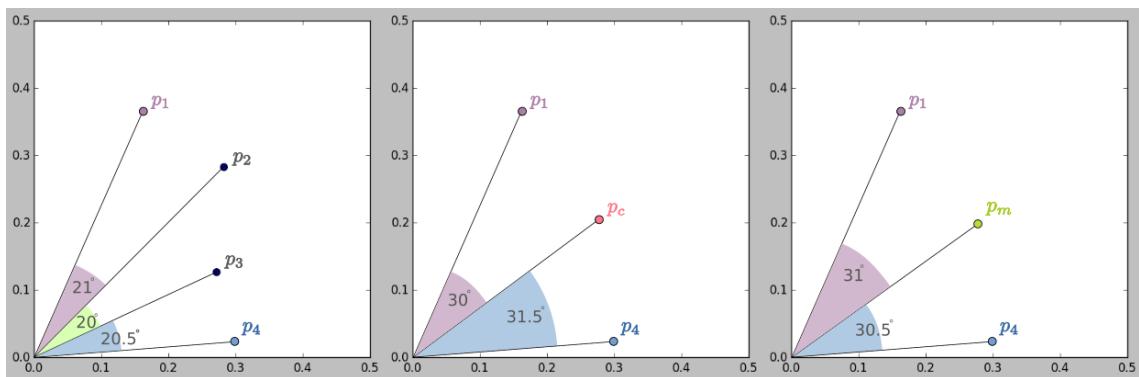


Fig. 4.4: El centroide ( $p_c$ ) de  $p_2$  y  $p_3$  no representa el punto medio ( $p_m$ ) respecto al ángulo entre ellos.

Podemos apreciar que al principio  $d(p_2, p_3) < d(p_3, p_4) < d(p_1, p_2)$ , siendo  $d(x, y)$  la distancia coseno (ecuación 3.2). Sin embargo, luego de utilizar el *linkage centroid* (ecuación 3.3) sucede que  $d(p_1, p_c) < d(p_4, p_c)$ .  $p_4$  es ahora el punto que mas lejos está del centroide. Creando un representante  $p_m$  usando el ángulo medio entre  $p_2$  y  $p_3$  esto no sucede. Este fenómeno se da porque la distancia coseno tiene en cuenta el ángulo pero el centroide no. Por lo tanto el centroide no caracteriza al punto medio respecto a la distancia coseno.

#### 4.2.3. Complejidad algorítmica del clustering

El diagrama de la figura 3.1 divide el proceso de parcelación de la corteza en varios pasos: seleccionar los voxels que serán utilizados como semilla de cada tractograma; luego generar cada tractograma y finalmente agruparlos usando algún algoritmo de clustering. En todo este proceso el paso mas caro en términos computacionales es el *clustering*. Veamos la complejidad del método propuesto por Moreno-Dominguez. En cada iteración del algoritmo es necesario calcular un centroide, almacenarlo y compararlo explícitamente con el resto de los clusters. Por cada iteración es necesario hacer  $O(c^2m)$  operaciones para recalcular todas las distancias, donde  $c$  es la cantidad de clusters y  $m$  es la longitud de los mismos. Dadas  $s$  semillas iniciales, la cantidad de iteraciones a realizar son  $s - 1$ . La complejidad temporal de este método es  $O(s^3m)$ . Respecto a la complejidad espacial, el mantener todos los clusters en memoria requiere  $O(sm)$  espacio. También es necesario mantener la matriz de distancias entre *clusters*, la cual requiere  $O(s^2)$ . La complejidad espacial total es  $O(sm + s^2)$ . Podemos notar que ambas complejidades dependen de  $m$ .

Resumiendo, utilizar *Hierarchical Agglomerative Clustering* con la distancia coseno y el *linkage* centroide presenta algunas desventajas: La medida de similitud y el *linkage* no son compatibles; no permite agrupar de manera correcta clusters colineales y necesita mantener todos los *clusters* en memoria así como también compararlos explícitamente.

### 4.3. Transformando los tractogramas al espacio *LogOdds*

En la sección anterior presentamos algunos de los inconvenientes teóricos del método propuesto por Moreno-Dominguez. Los mismos eran el clustering de vectores colineales; la relación entre la métrica y el *linkage* y la alta complejidad algorítmica. En esta sección proponemos una forma de solucionarlos haciendo uso de la función *logit*. La función *logit* permite convertir vectores provenientes de distribuciones discretas al espacio *LogOdds*. El espacio *LogOdds* es un espacio vectorial. Esto quiere decir que es cerrado respecto a la suma y la multiplicación por escalares. A continuación presentamos la transformación *logit* en detalle; luego mostramos como esto permite solventar los problemas presentados y mejorar la complejidad algorítmica del *clustering*.

#### 4.3.1. Transformación Logit

Sea  $P_M$  el espacio de una distribución discreta para  $M$  etiquetas:

$$P_M = \left\{ p \mid p = (p_1, \dots, p_n) \in (0, 1)^M, \sum p_i = 1 \right\}$$

La función  $\text{logit}: P_M \rightarrow R^{M-1}$  define una transformación entre el espacio  $P_M$  y un espacio vectorial  $R^{M-1}$ . A este espacio vectorial se lo denomina *LogOdds*. Dados los vectores  $Q \in P_M$  y  $S \in R^{M-1}$ :

$$S_i = \text{logit}(Q_i) = \log \left( \frac{Q_i}{Q_M} \right)$$

Para el caso de  $M = 2$  el espacio de origen es la distribución Bernoulli discreta. Podemos ver en la ecuación 4.1 su expresión analítica y en la figura 4.5 su representación gráfica.

$$\text{logit}(p) = \log \left( \frac{p}{1-p} \right) \quad (4.1)$$

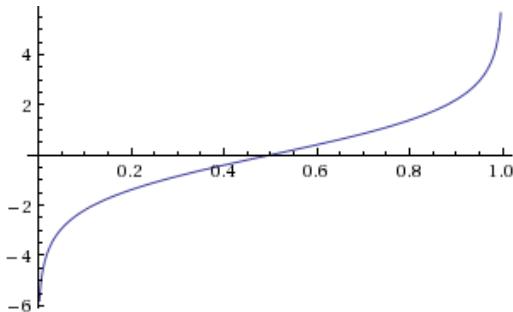


Fig. 4.5: Representación gráfica de la función *logit*. La misma nos permite transformar los tractogramas a un espacio vectorial.

Trabajar en un espacio vectorial nos asegura que la suma y la multiplicación por escalares están contenidos en el mismo espacio. Pohl et al. [37] utilizan esta propiedad para realizar operaciones lineales sobre mapas probabilísticos. A su vez, muestran que la suma y multiplicación por escalares en el espacio *LogOdds* poseen un significado en el espacio de la distribución. La suma en el espacio *LogOdds* representa una multiplicación normalizada en el espacio de origen:

$$p_1 \oplus p_2 = \text{logit}^{-1}(\text{logit}(p_1) + \text{logit}(p_2)) = \frac{1}{p_1 \cdot p_2} (p_{11}p_{21}, \dots, p_{1n}p_{2n})$$

donde  $p_1$  y  $p_2$  son vectores provenientes de una misma distribución y  $X_1 \cdot X_2$  es el producto interno entre  $X_1$  y  $X_2$ . Multiplicar por un escalar  $\alpha$  en el espacio de *LogOdds* equivale a exponenciar la distribución por  $\alpha$  y normalizarla:

$$\alpha \otimes p = \text{logit}^{-1}(\text{logit}(p_1)\alpha) = \frac{1}{\sum p_i^\alpha} (p_1^\alpha, \dots, p_n^\alpha)$$

#### 4.3.2. Modificando el algoritmo de clustering

Veamos como modificar el agrupamiento de tractogramas utilizando la función *logit*. Asumiendo que cada voxel  $v$  de un tractograma proviene de la variable aleatoria binaria:

$$X_v = \text{"La semilla está conectada con el voxel } v\text{"}$$

Es posible transformar el tractograma aplicando la función *logit* en cada uno de sus voxels. El resultado es un vector donde cada coordenada se encuentra en el espacio *LogOdds*. Más aún, las operaciones lineales entre los mismos voxels en distintos tractogramas están definidas. Esto nos permite usar la métrica euclíadiana como función de similitud en *Agglomerative Hierarchical Clustering*.

Transformar de espacio los tractogramas y cambiar la función de similitud posee varias ventajas. Para empezar permite agrupar correctamente vectores colineales. La figura 4.6 muestra el resultado de aplicar este método a los vectores de la figura 4.1. Podemos apreciar que las tres poblaciones se encuentran correctamente separadas y bien definidas.

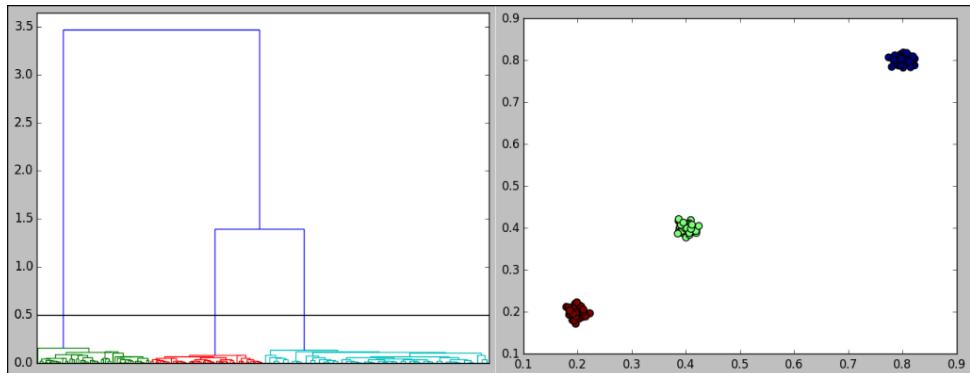


Fig. 4.6: Clustering resultado de utilizar nuestro método para agrupar los vectores. Los tres clusters fueron separados correctamente.

Otra ventaja es la buena relación métrica-*linkage*. Por definición el centroide es el centro de masa de los clusters. Esto quiere decir que es el punto que minimiza la distancia euclíadiana entre los clusters que lo componen. Por ende, el centroide caracteriza bien el punto medio de los vectores en el espacio euclídeo.

Finalmente, nuestro método también permite mejorar la complejidad algorítmica. El algoritmo *Agglomerative Hierarchical Clustering* une en cada iteración dos *clusters*, calcula un representante de estos y luego computa su distancia al resto. Usar la métrica euclíadiana junto con el *linkage* centroide es posible simplificar estos pasos.

$$d(i \cup j, k) = \alpha_i d(i, k) + \alpha_j d(j, k) + \beta d(i, j) + \gamma |d(i, k) - d(j, k)| \quad (4.2)$$

$$\alpha_i = \frac{|i|}{|i| + |j|} \quad \alpha_j = \frac{|j|}{|i| + |j|} \quad \beta = -\frac{|k|}{|i| + |j| + |k|} \quad \gamma = 0 \quad (4.3)$$

La formula de Lance y Williams (ecuación 4.2) permite computar las distancias entre *clusters* sin compararlos explícitamente. Para el método centroide los parámetros son las relaciones de la ecuación 4.3. Usar esta formula reduce significativamente la complejidad. Cada iteración pasa a costar  $O(c^2)$  en vez de  $O(c^2m)$ , siendo  $c$  la cantidad de clusters y  $m$  la longitud de los mismos. Dadas  $s$  semillas iniciales, la complejidad temporal total del *clustering* es  $O(s^3)$ . En la sección 4.2.3 se ve que usando la distancia coseno era  $O(s^3m)$ , siendo  $m$  la longitud de los tractogramas. El no comparar explícitamente los clusters permite no tener que mantenerlos en memoria. Esto reduce significativamente la complejidad temporal. Solo es necesario almacenar la matriz de distancias entre clusters. Esto tiene un costo espacial total de  $O(s^2)$ . La sección 4.2.3 muestra que usando la distancia coseno esta complejidad era  $O(sm + s^2)$ . Recordemos que en el contexto en que estamos utilizando este algoritmo  $m \gg s$ . Por lo tanto, este resultado implica una gran reducción en la complejidad del algoritmo.

#### 4.3.3. Almacenando tractogramas: Matrices ralas

Los métodos para parcelar la corteza utilizados en este trabajo están basados en el *clustering* de tractogramas. Podemos ver en la figura 3.1 que primero generamos los tractogramas y luego se los damos a un algoritmo para que los agrupe. Entre estos pasos es necesario almacenar o mantener en memoria los tractogramas. En esta sección veremos como aprovechar la propiedad rala de los tractogramas para almacenarlos eficientemente. También mostramos como recuperar esta propiedad luego de transformarlos al espacio *LogOdds*.

A modo de ejemplo situamos 762 semillas en el área de Broca de un sujeto y generamos sus tractogramas. Aplanamos todos los tractogramas a una sola dimensión y los usamos como filas de una matriz. La matriz resultante tiene dimensiones  $762 \times 3587328$ . Asumiendo que cada valor se representa usando 8 Bytes (*double precision*), esta matriz ocupa un total de aproximadamente 20 Gigabytes. Sin embargo, solo un 1% de los datos almacenados son no nulos. Esto implica que casi todo el espacio utilizado es desperdiciado.

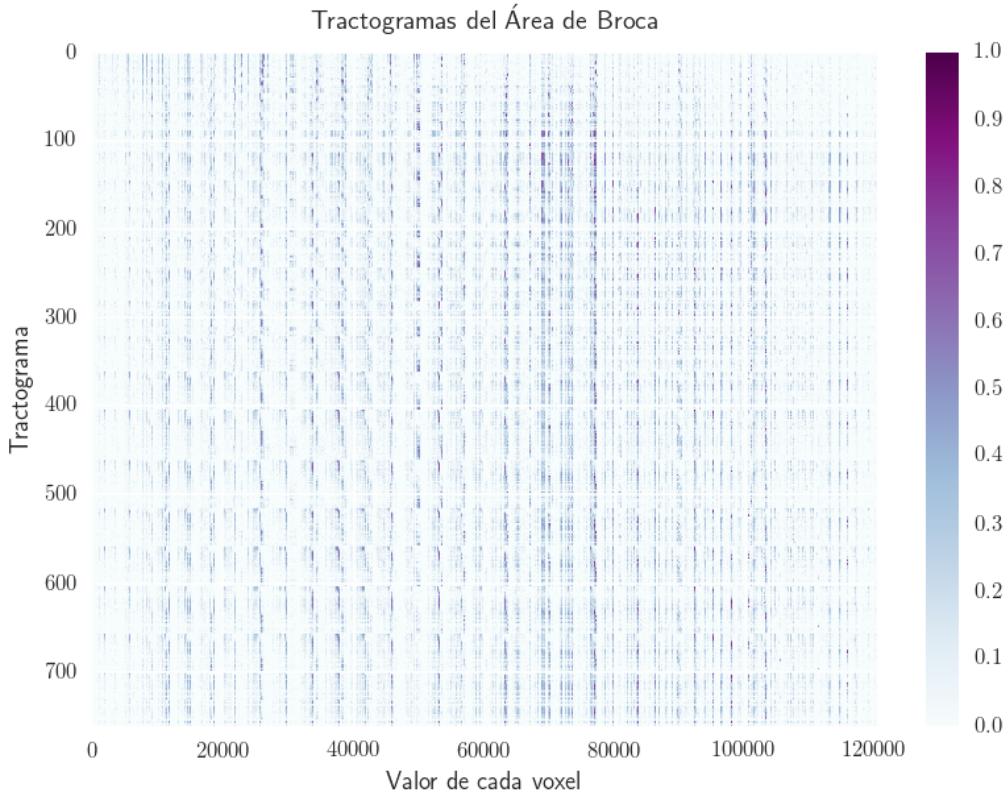


Fig. 4.7: Matriz con los tractogramas generados en el área de Broca de un sujeto. Las columnas cuyos valores eran todos nulos fueron removidas.

Una forma de reducir el espacio requerido para almacenar la matriz es eliminar las columnas con solo elementos nulos. La figura 4.7 muestra la matriz que resulta de eliminar las columnas vacías de la matriz del área de Broca. Manteniendo la representación de 8 Bytes este proceso reduce el espacio necesario a aproximadamente 700 Megabytes. Si bien esto es una gran mejora, la nueva matriz posee solo un 27 % de valores no nulos. Seguimos desperdi ciando mucho espacio.

En el caso de crear los tractogramas de todo el hemisferio derecho usando 21657 semillas el eliminar columnas no es suficiente. La matriz resultante posee dimensiones  $21657 \times 3587328$ . En esta, solo un 1 % de los valores son no nulos. Para almacenar dicha matriz es necesario utilizar 587 Gigabytes. Por esto es necesario utilizar estructuras mas eficiente, que aprovechen lo ralo de las matrices. Ejemplos de estas estructuras son: *Dictionary of Keys*, o una matriz *Compressed Sparse Row* (CSR).

Al momento querer aplicar esto en nuestro método nos encontramos con un inconveniente. Nosotros transformamos los tractogramas al espacio *LogOdds* antes de agruparlos. Por la forma que tiene la función *logit* (ecuación 4.1) los tractogramas transformados ya no son ralos. Podemos ver en la figura 4.5 que  $\text{logit}(0) = -\infty$ . Esto implica que la transformación de una matriz rala no es rala en términos de elementos nulos. Sin embargo podemos aprovechar ciertas propiedades para recuperar las matrices ralas. La distancia euclíadiana entre vectores es invariante a traslaciones lineales del sistema. Lo mismo sucede con las

posiciones relativas de los *centroïdes*. Asignemos una representación finita  $c$  al valor  $-\infty$ . Un buen candidato para  $c$  es el  $\log(\epsilon)$ , donde  $\epsilon$  es el *epsilon de la maquina*. Transformar todos los vectores y luego trasladarlos sumando  $c$  en cada componente dará como resultado una representación rala. Gracias a esto podemos utilizar DOK, CSR o cualquier estructura para reducir el espacio necesario para almacenar los tractogramas.

Dadas dos matrices ralas existen maneras eficientes de multiplicarlas entre si [38]. Si calculamos la distancia euclíadiana entre dos vectores ralos usando la siguiente relación:

$$\text{simil}_{\text{euc}}(X, Y) = \sqrt{X \cdot X^t - 2X \cdot Y^t + Y \cdot Y^t} \quad (4.4)$$

Podemos generar la matriz de distancias usando directamente las estructuras de matrices ralas. Es importante destacar que este método generara una matriz que no necesariamente es simétrica. Algunos valores que deberían ser iguales pueden presentar un pequeño error numérico. Esto nos permite utilizar matrices ralas durante todo el proceso de parcelamiento de la corteza.

#### 4.4. Parcelando la corteza en el espacio *LogOdds*

En la sección anterior presentamos las ventajas de transformar los tractogramas a un espacio vectorial. Mostramos que es posible realizar *clustering* con mejor complejidad espacial y temporal que el método Moreno-Dominguez. Ahora nos proponemos explicar todos los pasos que seguimos para parcelar la corteza de un sujeto. También presentamos el algoritmo de clustering en su totalidad. Recordemos del capítulo 3 que primero es necesario adquirir los datos de un sujeto; luego seleccionar los voxels que serán utilizados como semilla de cada tractograma; generar un tractograma por cada semilla y finalmente agruparlos usando algún algoritmo de clustering.

##### 4.4.1. Pre-clustering

Descargamos los datos de una mujer diestra de entre 23 y 26 años de *The Human Connectome Project* [18]. Usando el método descrito en la sección 3.2 situamos semillas en la materia blanca a 3mm de la corteza. Por cada semilla creamos un tractograma usando quince mil partículas. Los tractogramas fueron generados con el algoritmo 3 presentado en la sección . Descartamos aquellos tractogramas que poseían solo un voxel visitado. A su vez, los voxels de cada tractograma con valor menor a 0,25 fueron transformados en cero. Recordemos que los tractogramas provienen de la ecuación 3.1. Esto implica que dichos voxels fueron visitados por menos del 0,006 % de las partículas.

Cada tractograma fue transformado al espacio euclídeo utilizando la función *logit* presentada en la sección 6.4 y 4.3.2. Luego todos fueron trasladados linealmente como explicamos en la sección 4.3.3. Usando cada tractograma como una fila creamos una matriz CSR y calculamos la matriz de distancias entre ellos mediante la ecuación 4.4. Finalmente, definimos como vecinos a los voxels que se encuentran a una distancia menor a 3mm entre si.

**Algoritmo 5** Parcelamiento de la corteza cerebral

---

```

1: procedure PARCELAMIENTO( $K$ : PRIMEROS PASOS DEL CLUSTERING,
2:                               SEMILLAS: SEMILLAS,
3:                               dMRI:IMG. DE DIFUSIÓN,
4:                               ANATOMICA: IMG. ANATÓMICA)
5:    $n\_semillas \leftarrow \text{len}(semillas)$ 
6:    $tam\_anatomica \leftarrow \text{size}(anatomica)$ 
7:    $rala\_tractos \leftarrow \text{matriz rala de } n\_semillas \times tam\_anatomica$ 
8:    $infinito\_neg \leftarrow \text{logit}(\epsilon)$ 
9:   for  $S$  in  $semillas$  do
10:     $tracto\_actual \leftarrow \text{tractograma}(dMRI, anatomica, S)$ 
11:     $tracto\_logit \leftarrow \text{map(logit, tracto\_actual)}$ 
12:     $tracto\_logit\_traslacion \leftarrow tracto\_logit - infinito\_neg$ 
13:     $rala\_tractos \leftarrow \text{agregar tracto\_logit\_traslacion como fila}$ 
14:     $vecinos \leftarrow \text{matriz cuadrada de dimensiones } n\_semillas \times n\_semillas$ 
15:     $distancias \leftarrow \text{matriz cuadrada de dimensiones } n\_semillas \times n\_semillas$ 
16:     $vecinos \leftarrow semillas$  que estan fisicamente a cierta distancia
17:     $distancias \leftarrow \text{distancia euclidea entre todos los pares de } semillas$ 
18:     $clusters \leftarrow \text{CLUSTERING}(vecinos, distancias)$ 
19:   return  $clusters$ 

```

---

El algoritmo 5 muestra y resume lo dicho en forma de pseudocódigo. Este proceso de creación de estructuras posee la misma complejidad tanto en nuestro método como en el de Moreno-Dominguez.

#### 4.4.2. Algoritmo de clustering

Comparando nuestro algoritmo de *clustering* (algoritmo 6) con el de Moreno-Dominguez (algoritmo 2) se pueden ver explícitamente las diferencias. A continuación las detallamos mostrando como se traducen en una mayor eficiencia espacial y temporal.

El método de Moreno-Dominguez necesita mantener en memoria los tractogramas durante el clustering. En nuestro algoritmo basta con tener la matriz de distancia para generar la jerarquía. Por lo visto en la sección 4.3.3 esto implica un ahorro significativo de memoria. Luego, al momento de calcular las distancias al nuevo centroide (línea 15, algoritmo 6) usamos la fórmula de Lance y Williams (ecuación 4.2). Esta fórmula nos permite mejorar aún más la complejidad algorítmica. A continuación repetimos los resultados del análisis de complejidad hecho en la sección 4.3.2. Suponiendo un número  $s$  de semillas y una cantidad  $m$  de voxels por cada tractograma. Nuestro algoritmo de clustering posee complejidad espacial  $O(s^2)$  contra  $O(s^2 + sm)$  del método Moreno-Dominguez. A su vez, la complejidad temporal de nuestro algoritmo es  $O(s^2m)$  contra  $O(s^2)$  del método Moreno-Dominguez. Recordemos de la sección 4.3.3 que dentro del contexto de este trabajo  $m >> s$ . La mejora en complejidad algorítmica es significativa.

Una vez finalizado el proceso de *clustering* obtenemos como resultado un dendrograma. Para parcelar la corteza solo es necesario seleccionar una altura en la cual cortar dicho

---

**Algoritmo 6** Clustering de tractogramas

---

```

1: procedure CLUSTERING(K_PASOS: PRIMEROS K PASOS,
2:                               VECINOS: MAT. DE VECINOS,
3:                               DISTANCIAS: MAT. DE DISTANCIA CLUSTERS )
4:   jerarquia  $\leftarrow \emptyset$ 
5:   clusters  $\leftarrow [1\dots\text{len}(\text{distancias})]$ 
6:   for k in  $[1, \text{len}(\text{distancias})]$  do
7:     if k > k_pasos then
8:        $C_x, C_y \leftarrow \text{clusters}$  tales que x e y poseen distancia minima
9:     else
10:       $C_x, C_y \leftarrow x$  e y son vecinos; de distancia minima y de tamaño similar.
11:      clusters  $\leftarrow$  eliminar clusters  $C_x, C_y$ 
12:      clusters  $\leftarrow$  agregar el numero del centroide  $(C_x, C_y)$ 
13:      jerarquia  $\leftarrow$  agregar la union  $(C_x, C_y)$ 
14:      for  $C_z$  in clusters do
15:        D  $\leftarrow$  distancia de  $C_z$  al centroide usando la formula Lance y Williams
16:        distancias  $\leftarrow$  actualizar distancia entre  $(C_x, C_y)$  y  $C_z$  con D
17:   return jerarquia

```

---

dendrograma. Los clusters que estén por debajo de ese corte serán las distintas parcelas.

Podemos ver que transformar los tractogramas utilizando la función *logit* presenta varias ventajas. Como vimos en la sección anterior las operaciones lineales están cerradas en el espacio *LogOdds*. Esto nos permite, calcular el promedio de un grupo de tractogramas. También mostramos que usar como medida de similitud la distancia euclídea mejora el *clustering*: Los *clusters* colineales son agrupados correctamente y la medida de similitud es consistente con el *linkage* centroide. A su vez, vimos como es posible almacenar los tractogramas transformados en matrices ralas. Finalmente, todo esto nos permite bajar en **al menos un orden la complejidad temporal y espacial**. No necesitamos mantener en memoria los tractogramas durante el *clustering* y podemos utilizar la formula de Lance y Williams para calcular la distancia entre *clusters* de manera más rápida.

## 5. RESULTADOS

En el capítulo 3 presentamos la técnica de *bootstraping*; el método de Moreno-Dominguez para agrupar tractogramas junto con algunas de sus falencias teóricas y como solucionarlas usando la función *logit*. En la sección 6.2, presentamos nuestro método para parcelar la corteza en su totalidad. En las siguientes secciones mostramos primero los resultados obtenidos al estudiar la estabilidad de los tractogramas; luego parcelamos el área de Broca con ambos métodos y finalmente parcelamos el hemisferio derecho usando también ambos métodos. Todos los estudios se realizaron sobre una misma mujer diestra de entre 23 y 26 años. Sus datos fueron descargados de la base de datos *Human Connectome Project* [18].

### 5.1. Convergencia de los tractogramas

El algoritmo 3 de la sección 5.3 genera tractogramas mediante un procedimiento Monte Carlo. Los tractogramas creados de esta manera son inherentemente estocásticos. Queremos analizar si dado un número suficiente de partículas el resultado del algoritmo converge. Para ello utilizamos la técnica estadística de *bootstrap*. Creamos una población de *streamlines* y luego estudiamos como se comportan la media y la varianza de los tractogramas. El proceso es el detallado en la sección 3.3. Las Figuras 5.1, 5.3 y 5.5 muestran, para tres semillas distintas, cinco cortes coronales del tractograma creado usando 15000 *streamlines*. Las figuras 5.2, 5.4 y 5.6 muestran la varianza de cada voxel dentro de un mismo corte coronal entre tractogramas creados con distintas cantidades de submuestras.

La figura 5.7 muestra la media y varianza de los voxels *A*, *B* y *C* marcados en las figuras 5.2, 5.4 y 5.6. Estos voxels fueron los que mayor varianza presentaron al generar tractogramas usando subpoblaciones de 2000 *streamlines*.

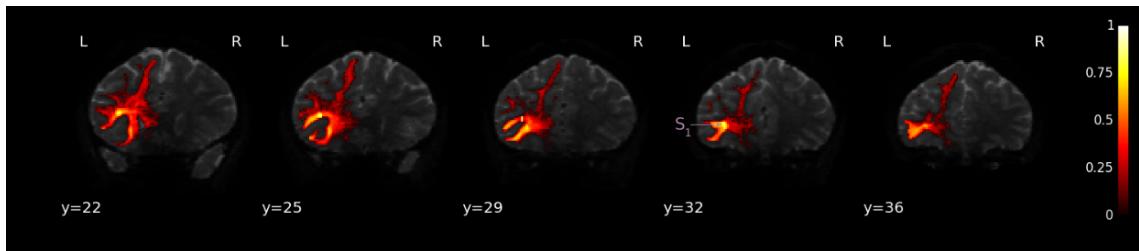


Fig. 5.1: Tractograma generado desde la semilla  $S_1$  utilizando 15000 *streamlines*.

### 5.2. Parcelamiento del área de Broca

Decidimos comenzar parcelando solo el área de Broca. El ser una región chica permite generar resultados en poco tiempo y comparar de manera visual los métodos implementados. Situamos semillas a 3mm de la corteza siguiendo la sección 3.2 y generamos tractogramas utilizando el algoritmo 3 de la sección . En total se crearon aproximadamente 760 semillas

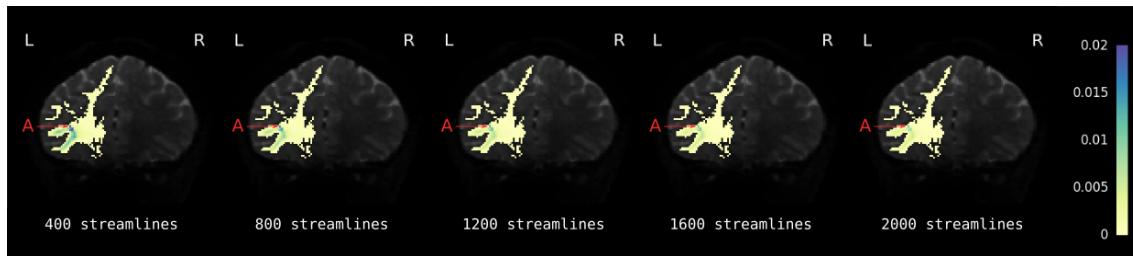


Fig. 5.2: Desviación estándar en un corte axial de los tractogramas de la semilla  $S_1$  variando la cantidad de streamlines.

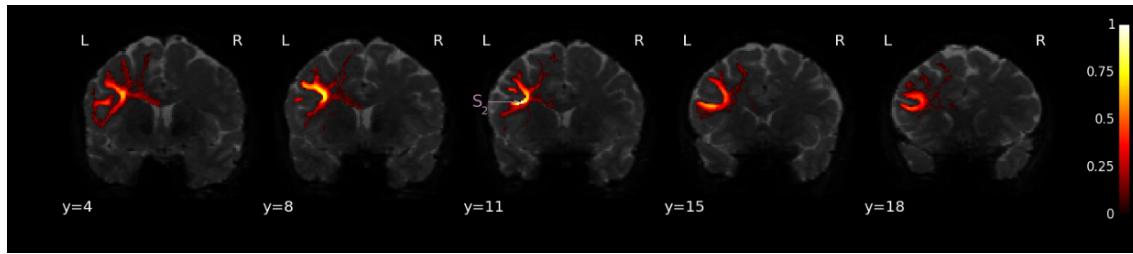


Fig. 5.3: Tractograma generado desde la semilla  $S_2$  utilizando 15000 streamlines.

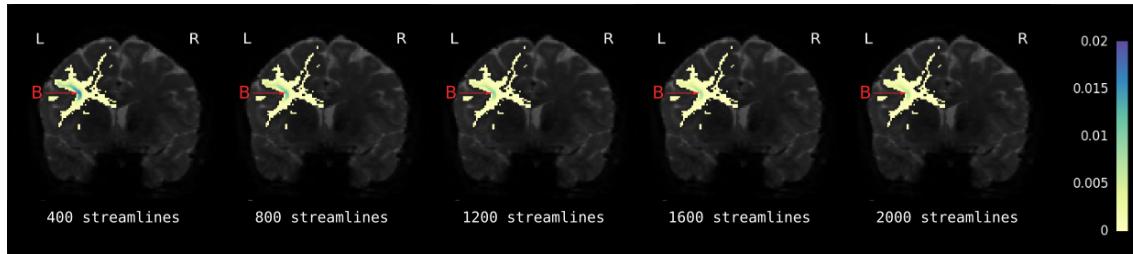


Fig. 5.4: Desviación estándar en un corte axial de los tractogramas de la semilla  $S_2$  variando la cantidad de streamlines.

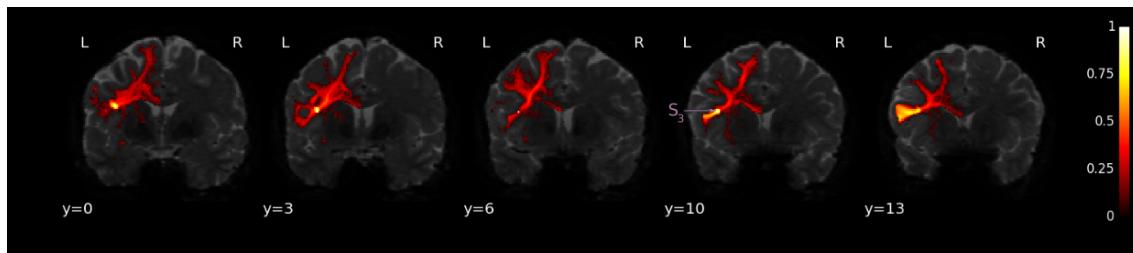


Fig. 5.5: Tractograma generado desde la semilla  $S_3$  utilizando 15000 streamlines.

con sus tractogramas. A continuación mostramos distintas parcelaciones de la corteza obtenidas mediante el método de Moreno-Dominguez y el nuestro.

### 5.2.1. Método Moreno-Dominguez

Parcelamos el área de Broca usando el método de Moreno-Dominguez. Al igual que en la sección , descartamos los tractogramas con un solo voxel visitado y aplicamos un

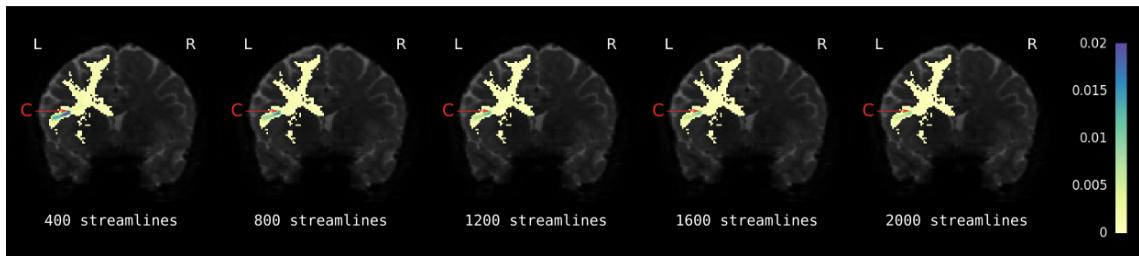


Fig. 5.6: Desviación estándar en un corte axial de los tractogramas de la semilla  $S_3$  variando la cantidad de streamlines.

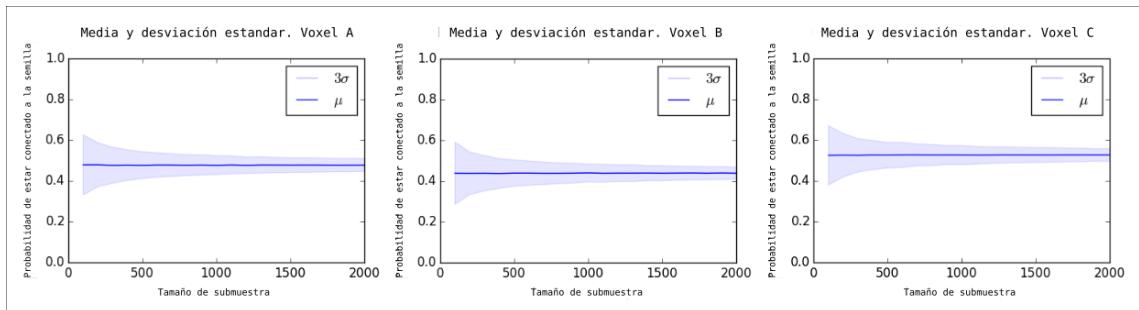


Fig. 5.7: Media y desviación estándar de los voxels A, B y C. Estos fueron los voxels con mayor varianza entre tractogramas generados con 2000 streamlines.

*threshold* de 0,4 en cada voxels. Las figuras 5.8 y 5.9 se generaron sin aplicar restricciones en ninguna iteración del *clustering* ( $k = 0$ ). Ambas muestran el resultado obtenido de un mismo dendrograma cortado a distintos niveles de profundidad. La figura 5.10 muestra los resultados usando  $k = 400$ . Esto es, las primeras 400 uniones son solo entre *clusters* vecinos y de tamaño homogéneo. La figura 5.11 muestra la parcelación obtenida usando  $k = 700$ .

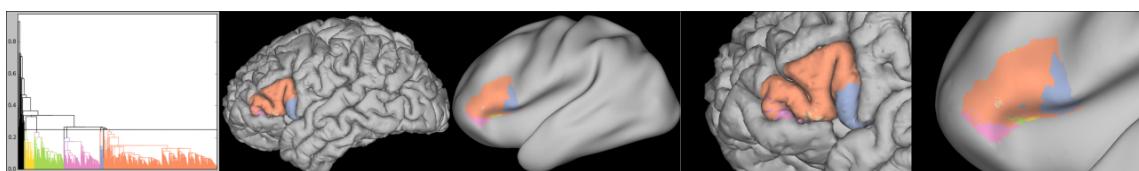


Fig. 5.8: Parcelamiento del área de Broca utilizando el método de Moreno-Dominguez sin aplicar restricciones en ninguna iteración.

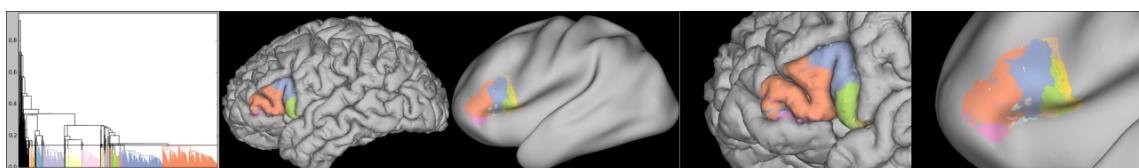


Fig. 5.9: Parcelamiento del área de Broca utilizando el método de Moreno-Dominguez sin aplicar restricciones en ninguna iteración. Corte con mayor profundidad en el dendrograma.

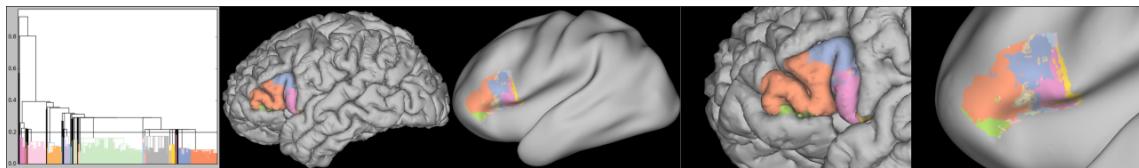


Fig. 5.10: Parcelamiento del área de Broca utilizando el método de Moreno-Dominguez. Las primeras 400 uniones fueron entre clusters vecinos.

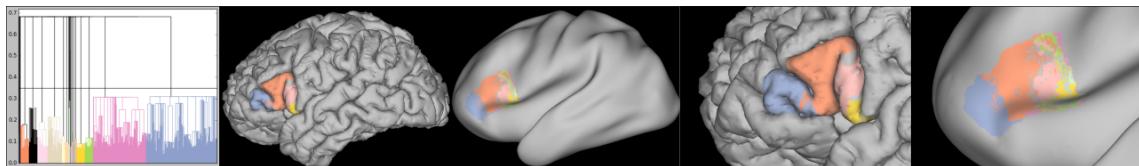


Fig. 5.11: Parcelamiento del área de Broca utilizando el método de Moreno-Dominguez. Las primeras 700 uniones fueron entre clusters vecinos.

### 5.2.2. Parcelamiento en el espacio LogOdds

Parcelamos el área de Broca usando nuestro método. Al igual que en la sección , descartamos los tractogramas con un solo voxel visitado y aplicamos un *threshold* de 0,25 en cada voxels. Las figuras 5.12 y 5.12 se generaron sin aplicar restricciones en ninguna iteración del *clustering* ( $k = 0$ ). Ambas muestran el resultado obtenido de un mismo dendrograma cortado a distintos niveles de profundidad. La figura 5.14 muestra los resultados usando  $k = 400$ . Esto es, las primeras 400 uniones son solo entre *clusters* vecinos y de tamaño homogéneo. La figura 5.15 muestra la parcelación obtenida usando  $k = 700$ .

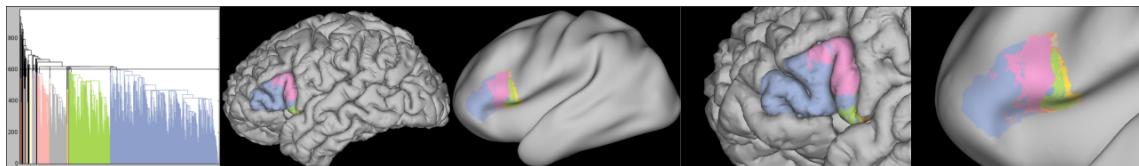


Fig. 5.12: Parcelamiento del área de Broca utilizando nuestro método sin aplicar restricciones en ninguna iteración.

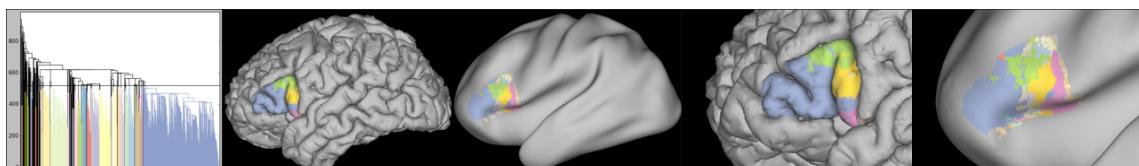


Fig. 5.13: Parcelamiento del área de Broca utilizando nuestro método sin aplicar restricciones en ninguna iteración. Corte con mayor profundidad en el dendrograma.

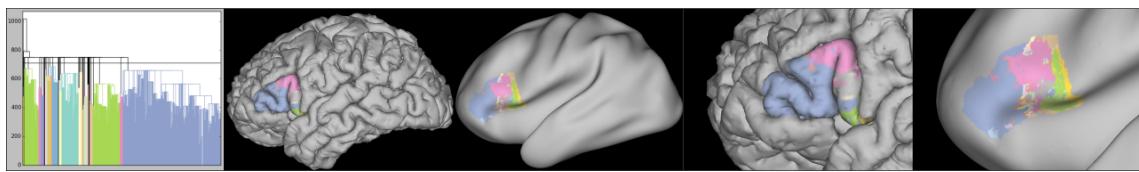


Fig. 5.14: Parcelamiento del área de Broca utilizando nuestro método. Las primeras 400 uniones fueron entre *clusters* vecinos.

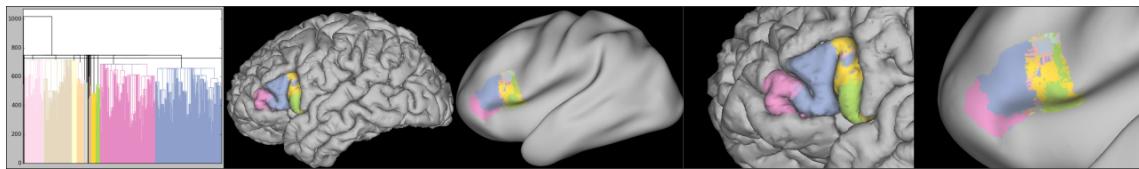


Fig. 5.15: Parcelamiento del área de Broca utilizando nuestro método. Las primeras 700 uniones fueron entre *clusters* vecinos.

### 5.2.3. Resultados de los métodos en mayor detalle

Para facilitar una primera comparación visual entre los métodos presentamos las parcelaciones con mayor detalle. Las figuras 5.16; 5.17 y 5.18 presentan parcelaciones obtenidas usando  $k = 0$ ,  $k = 400$  y  $k = 700$  respectivamente. Esto es: el algoritmo sin restricciones; las primeras 400 uniones entre *clusters* vecinos y las primeras 700 uniones entre *clusters* vecinos respectivamente. En cada figura se puede apreciar una parcelación obtenida con el método Moreno-Dominguez a la izquierda y una parcelación obtenida con nuestro método a la derecha. Los dendrogramas no se presentan, dada la diferencia de alturas y cortes no aportan a la comparación visual.

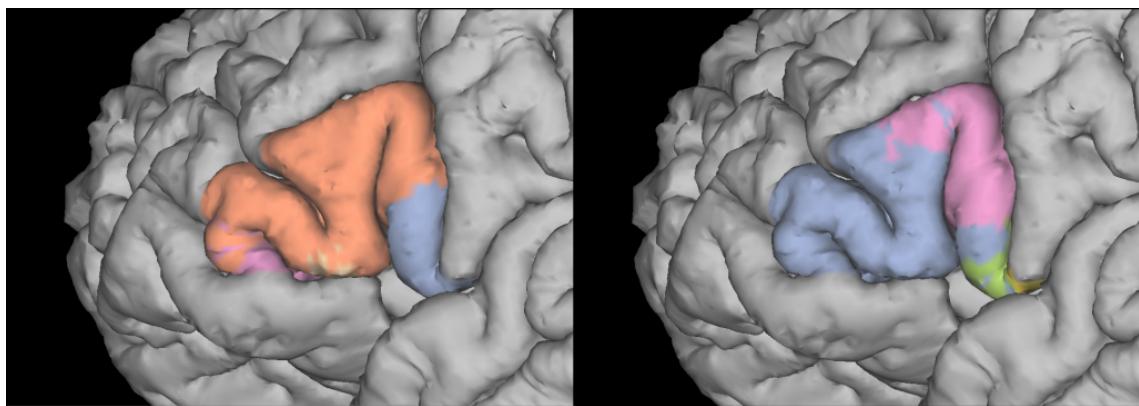


Fig. 5.16: Detalle sobre las parcelas obtenidos utilizando el método de Moreno-Dominguez (izquierda) y el nuestro (derecha). No se aplicaron restricciones en las iteraciones.

## 5.3. Parcelamiento del hemisferio izquierdo

Luego de dividir el área de Broca parcelamos todo el hemisferio izquierdo del sujeto. Nuevamente situamos semillas a 3mm de la corteza siguiendo la sección 3.2 y generamos

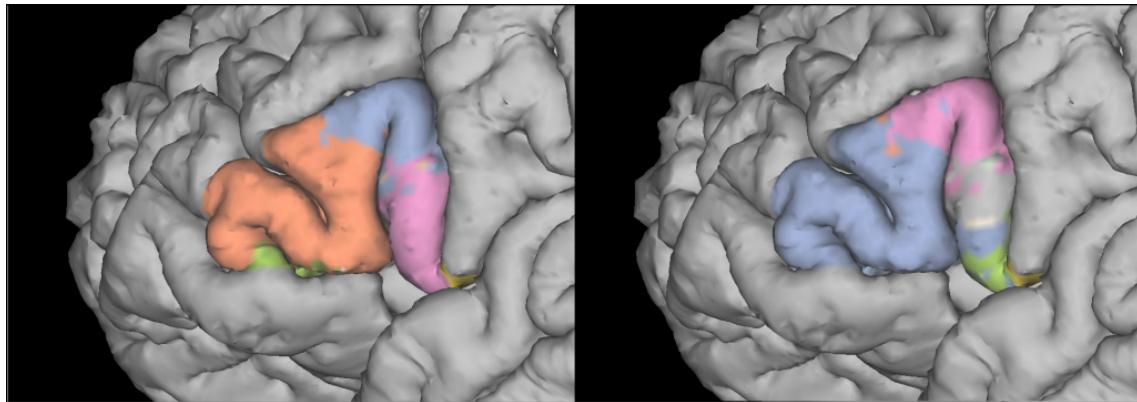


Fig. 5.17: Detalle sobre las parcelas obtenidos utilizando el método de Moreno-Dominguez (izquierda) y el nuestro (derecha). Las primeras 400 uniones fueron entre clusters vecinos.

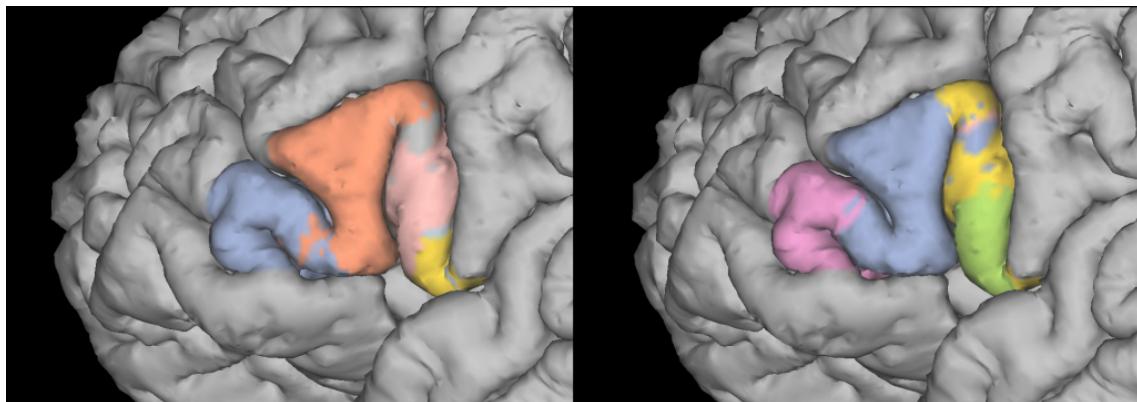


Fig. 5.18: Detalle sobre las parcelas obtenidos utilizando el método de Moreno-Dominguez (izquierda) y el nuestro (derecha). Las primeras 700 uniones fueron entre clusters vecinos.

tractogramas utilizando el algoritmo 3 de la sección . En este caso hubo aproximadamente 24000 semillas con sus respectivos tractogramas. A continuación mostramos distintas parcelaciones de la corteza obtenidas mediante el método de Moreno-Dominguez y el nuestro.

### 5.3.1. Método Moreno-Dominguez

Parcelamos el hemisferio usando el método de Moreno-Dominguez. Al igual que en la sección , descartamos los tractogramas con un solo voxel visitado y aplicamos un *threshold* de 0,4 en cada voxels. Las figuras 5.19; 5.20 y 5.21 muestran las parcelas obtenidas usando  $k = 10000$  a distintos niveles de profundidad del dendrograma. Esto es, las primeras 10000 uniones son solo entre *clusters* vecinos y de tamaño homogéneo.

### 5.3.2. Utilizando nuestro método

Por último parcelamos el hemisferio usando el método de Moreno-Dominguez. Al igual que en la sección , descartamos los tractogramas con un solo voxel visitado y aplicamos un

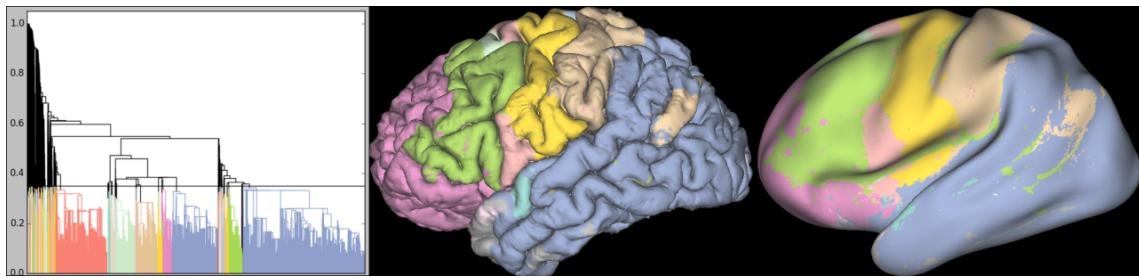


Fig. 5.19: Parcelamiento de la corteza utilizando el método de Moreno-Dominguez. Las primeras 10000 uniones fueron entre clusters vecinos.

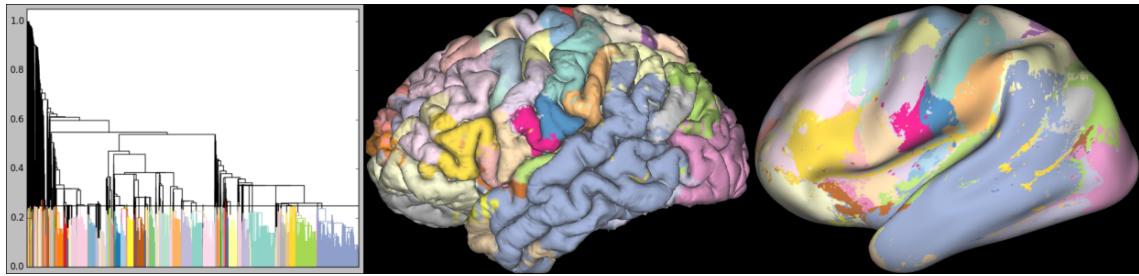


Fig. 5.20: Parcelamiento de la corteza utilizando el método de Moreno-Dominguez. Las primeras 10000 uniones fueron entre clusters vecinos. Corte con mayor profundidad en el dendrograma.

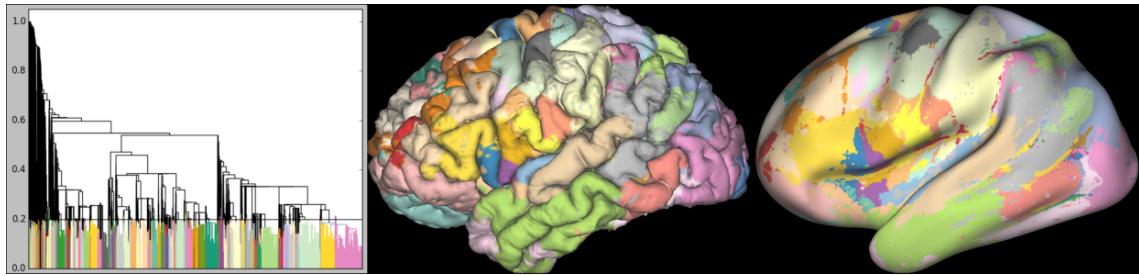


Fig. 5.21: Parcelamiento de la corteza utilizando el método de Moreno-Dominguez. Las primeras 10000 uniones fueron entre clusters vecinos. Corte con aún mayor profundidad en el dendrograma.

*threshold* de 0,25 en cada voxels. Las figuras 5.22 y 5.23 muestran las parcelas obtenidas sin aplicar restricciones en ninguna iteración del *clustering* ( $k = 0$ ). Ambas muestran el resultado obtenido de un mismo dendrograma cortado a distintos niveles de profundidad. Las figuras 5.24; 5.25 y 5.26 muestran los resultados usando  $k = 20000$  a distintos niveles de profundidad del dendrograma. Esto es, las primeras 10000 uniones son solo entre *clusters* vecinos y de tamaño homogéneo.

### 5.3.3. Resultados de los métodos en mayor detalle

A diferencia de lo que sucedía con el área de Broca, es difícil comparar los resultados visualmente. Sin embargo presentamos dos parcelaciones obtenidas para que puedan ser observadas en mayor detalle. La figura 5.27 presenta la corteza parcelada usando el método Moreno-Dominguez con  $k = 10000$ . Esto es, las primeras 10000 uniones son solo entre

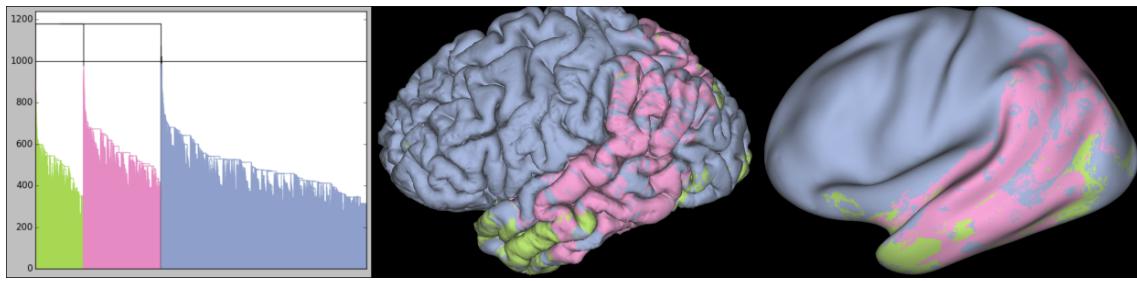


Fig. 5.22: Parcelamiento de la corteza utilizando nuestro método sin aplicar restricciones en ninguna iteración.

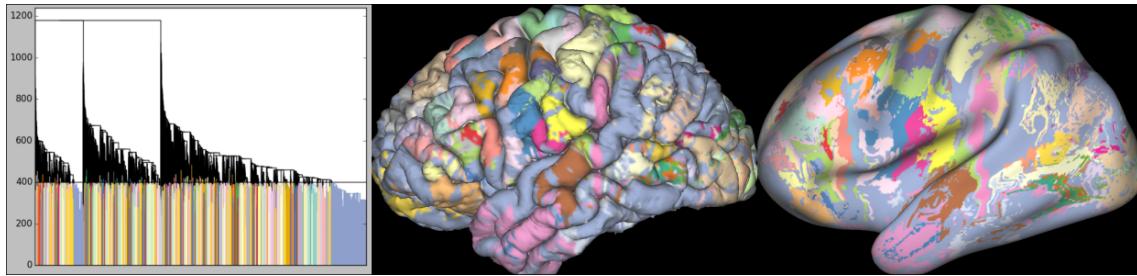


Fig. 5.23: Parcelamiento de la corteza utilizando nuestro método sin aplicar restricciones en ninguna iteración. Corte con mayor profundidad en el dendrograma.

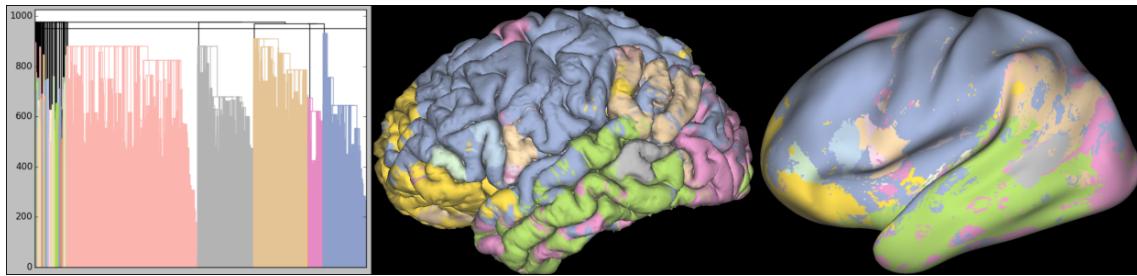


Fig. 5.24: Parcelamiento del área de Broca utilizando nuestro método. Las primeras 20000 uniones fueron entre clusters vecinos.

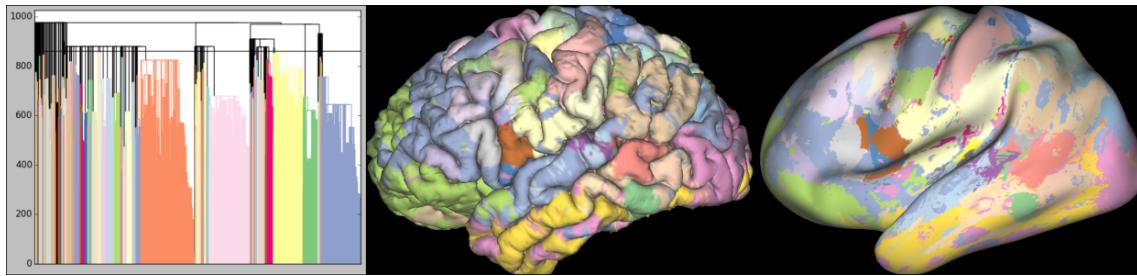


Fig. 5.25: Parcelamiento del área de Broca utilizando nuestro método. Las primeras 20000 uniones fueron entre clusters vecinos. Corte con mayor profundidad en el dendrograma.

clusters vecinos y de tamaño homogéneo. La figura 5.28 muestra el resultado de usar nuestro método con  $k = 20000$ .

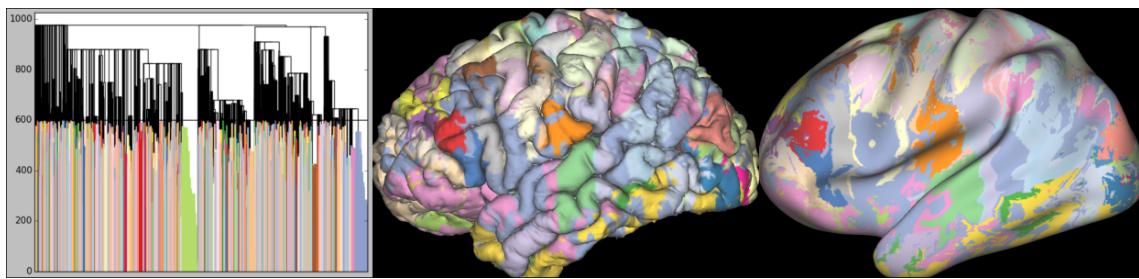


Fig. 5.26: Parcelamiento del área de Broca utilizando nuestro método. Las primeras 20000 uniones fueron entre clusters vecinos. Corte con aún mayor profundidad en el dendrograma.

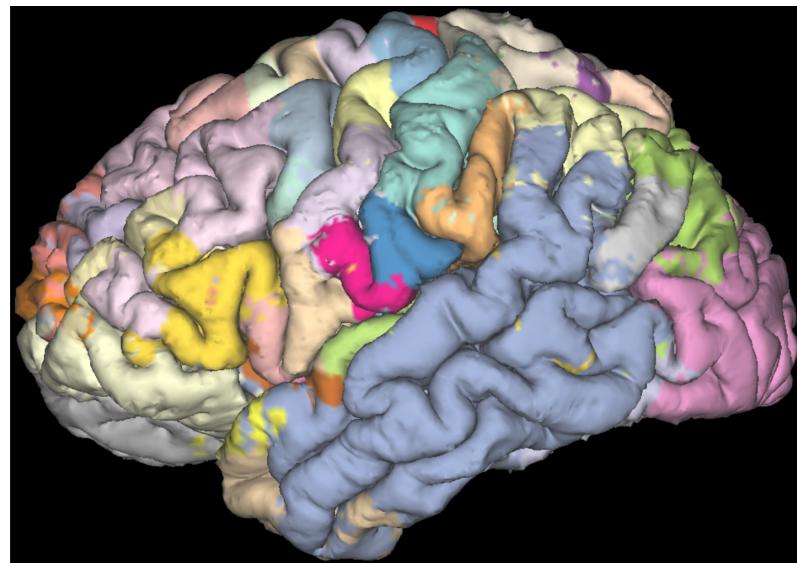


Fig. 5.27: Detalle sobre las parcelas obtenidos utilizando el método de Moreno-Dominguez. Las primeras 20000 uniones fueron entre clusters vecinos.

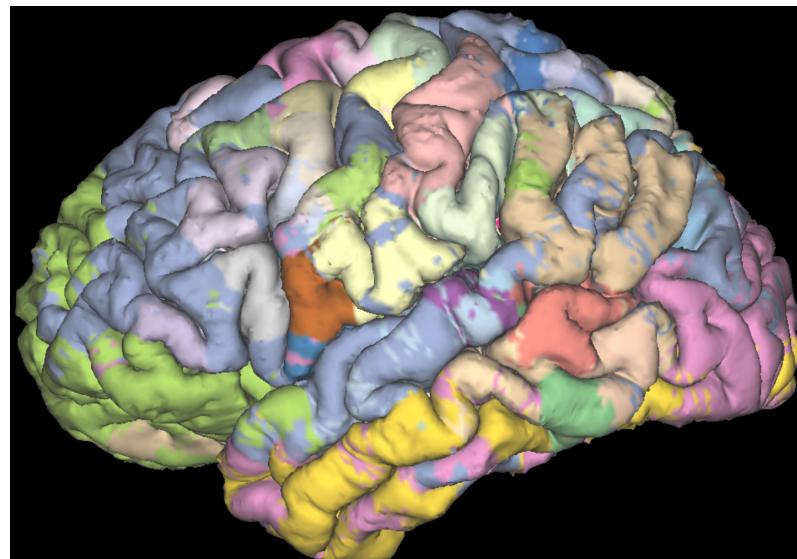


Fig. 5.28: Detalle sobre las parcelas obtenidos utilizando nuestro método. Las primeras 20000 uniones fueron entre clusters vecinos.



## 6. DISCUSIÓN

A lo largo de este trabajo presentamos como parcelar la corteza cerebral en base a un criterio estructural; estudiamos una de las métodos existentes actuales y presentamos un método propio. En los capítulos 1 y 2 introdujimos la parcelación estructural y sus fundamentos teóricos. En las secciones 5.3.1 y 4.2 analizamos el método de parcelamiento de Moreno-Dominguez [15]. El mismo se basa en combinar un algoritmo de tractografía y otro de *clustering* para parcelar la corteza cerebral. Luego, basándonos en su trabajo, propusimos en la sección un nuevo algoritmo de *clustering*. En este último capítulo vamos a discutir los resultados obtenidos. Comenzando por la robustez del algoritmo de tractografía utilizado; siguiendo con una comparación entre nuestro método y el de Moreno-Dominguez; continuando con la validez biológica de la parcelación obtenida y propondremos objetos de futuro estudio.

### 6.1. Convergencia del algoritmo de tractografía

Nuestro método de *clustering* se basa en agrupar tractogramas. Generamos cada tractograma usando el algoritmo 3 de la sección 5.3. Los tractogramas creados de esta manera son inherentemente estocásticos. Implementamos el algoritmo 4 de la sección 5.3 para estudiar si la creación de los tractogramas era estable. Esto es, si dado un número suficientemente grande de partículas el resultado converge a un único tractograma. Las figuras 5.2, 5.4 y 5.6 de la sección 5.1 muestran que la desviación estándar es casi nula al usar 15000 partículas. A su vez, la figura 5.7 muestra lo rápido que converge la media de tres voxels distintos. Como eran los de mayor desviación estándar, podemos asegurar que casi no existe diferencia en la media de los tractogramas creados con 2000 partículas. El algoritmo utilizado en el trabajo demostró ser estable.

Nuestro método de parcelamiento se basa en el clustering de tractogramas. Por ello es importante contar con un algoritmo de tractografía robusto. No encontramos estudios previos donde se analizara este aspecto del algoritmo de tractografía usado.

### 6.2. Comparación entre ambos métodos de *clustering*

En la sección 5.3.1 presentamos el método para parcelar la corteza de Moreno-Dominguez. Luego, en la sección presentamos nuestro método. El resultado de parcelar el área de Broca con estos se pueden ver en las secciones 5.2.1 y 5.2.2. En la sección presentamos los resultados lado a lado en detalle para compararlos. Los resultados son visualmente parecidos en todos los casos.

Luego utilizamos ambos algoritmo para parcelar el hemisferio izquierdo del mismo sujeto. Los resultados se encuentran en las secciones 5.3.1 y 5.3.2. En este caso resulta difícil comparar visualmente las parcelaciones. Más aún, es complicado encontrar cortes en los dendrogramas que generen representaciones parecidas. Dado que los arboles fueron

generados de manera distinta, sus cortes no tienen por que corresponderse. Sin embargo, la sección 5.3.3 muestra algunas similitudes en: la corteza motora y la somestésica; el lóbulo frontal; el lóbulo occipital y el área de Broca.

Respecto a la eficiencia computacional, en la sección 4.4.2 comparamos ambos métodos. Mostramos que nuestro método permite mejorar la complejidad temporal y espacial en al menos un orden de magnitud. Nuestro algoritmo de clustering posee complejidad espacial  $O(s^2)$  contra  $O(s^2 + sm)$  del método Moreno-Dominguez. La complejidad temporal de nuestro algoritmo es  $O(s^2m)$  contra  $O(s^2)$  del método Moreno-Dominguez. Recordemos de la sección 4.3.3 que dentro del contexto de este trabajo  $m >> s$ . En general el número de semillas usadas es muy menor a la cantidad de voxels de los tractogramas. En nuestro caso particular  $s \cong 22000$  y  $m \cong 3500000$ .  $m$  es 159 veces mas grande que  $s$ . Esto significa que la complejidad de nuestro método es significativamente menor a la del método de Moreno-Dominguez.

### 6.3. Correlato con otras parcelaciones de la literatura

Nuestro método divide la corteza cerebral basándose en el agrupamiento de tractogramas. El criterio usado es puramente estructural. Solo usamos información sobre la estructura física de la materia blanca. En la literatura actual existen parcelaciones basadas en otros criterios. Desikan et al. [39] dividen la corteza usando como referencia las circunvoluciones del cerebro. Peinfield [40] parcela la corteza en base a las respuestas a estímulos eléctricos sobre la misma. La parcelación de Desikan es anatómica, mientras que la de Peinfield es funcional. Una pregunta interesante es si nuestra parcelación posee correlación con otras anatómicas o funcionales. La relación entre el aspecto anatómico o funcional del cerebro y su aspecto estructural es un problema abierto en neurociencia. En el trabajo de Moreno-Dominguez solo hacen una breve comparación entre sus resultados y un mapa citoarquitectónico. A continuación contrastaremos nuestros resultados contra un atlas anatómico y un estudio funcional realizado sobre el sujeto.

#### 6.3.1. Correlación anatómica

En la sección 5.3.2 parcelamos el hemisferio izquierdo de un sujeto con nuestro método. Vamos a comparar el resultado obtenido con un atlas anatómico basado en Desikan et al. [39]. La figura 6.1 muestra el resultado de proyectar algunas regiones anatómicas sobre las obtenidas con nuestro método. La figura 6.2 muestra solo las regiones que estaban contenidas en su mayoría dentro de estas proyecciones. Luego calculamos que porcentaje de las parcelaciones quedaba dentro de las proyecciones; que porcentaje quedaba fuera y cuantas quedaban dentro de una sola proyección. El 90 % del área de las 9 proyecciones está pintada y solo un 5 % del área de nuestras parcelas queda fuera de las proyecciones. Las parcelas de nuestro método que se encuentran en una sola proyección representan el 88 % del área pintada. Las parcelas que están en dos regiones representan el 12 % restante. Esto indica que casi todas nuestras parcelas se encuentran contenidas dentro de a lo sumo una proyección. Nuestra parcelación creó áreas bien delimitadas dentro de al menos 9 regiones anatómicas.

Repetimos el procedimiento con los resultados obtenidos utilizando el método de Moreno-Dominguez. Proyectamos el atlas anatómico sobre la parcelación mostrada en la figura 5.21. En este caso el 75 % del área de las 9 proyecciones está pintada y un 15 % del área de las parcelas quedan fuera de las proyecciones. Sin embargo esto no quiere decir que el método de Moreno-Dominguez posea menor correlación con el atlas anatómico. La parcelación resultante en ambos métodos depende de la altura a la cual se decida cortar el dendrograma. Cortando el mismo dendrograma a mayor profundidad se pueden conseguir resultados similares a los de nuestro método.

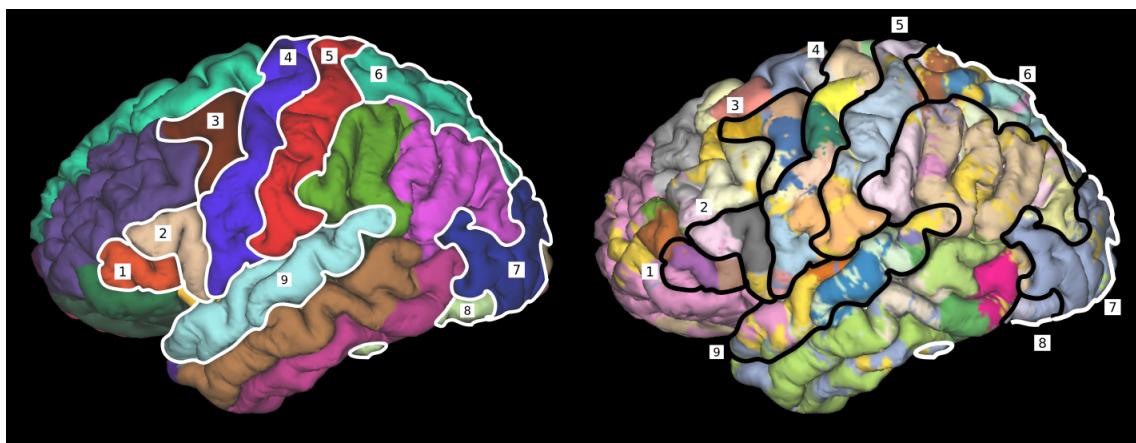


Fig. 6.1: Proyección del atlas anatómico (izquierda) a la parcelación obtenida (derecha). *Regiones:* 1. pars triangularis; 2. pars opercularis; 3. caudal middle frontal; 4. precentral (área motora); 5. post central; 6. superior parietal; 7. lateral occipital; 8. fusiform; 9. superior temporal.

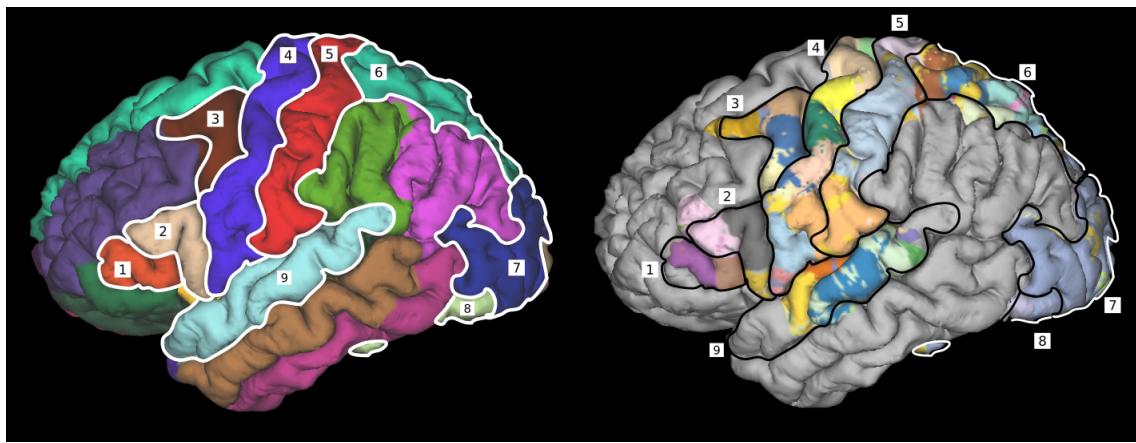


Fig. 6.2: Proyección del atlas anatómico (izquierda) a la parcelación obtenida (derecha). Solo las parcelas contenidas en mayor proporción fueron pintadas. *Regiones:* 1. pars triangularis; 2. pars opercularis; 3. caudal middle frontal; 4. precentral (área motora); 5. post central; 6. superior parietal; 7. lateral occipital; 8. fusiform; 9. superior temporal.

### 6.3.2. Correlación funcional

El sujeto de *Human Connectome Project* utilizado cuenta con un estudio funcional hecho por Bach et al. [3]. En el mismo utilizan Resonancia Magnética Funcional (fMRI) para estudiar la respuesta sobre la corteza a estímulos particulares. La Resonancia Magnética Funcional es un tipo especial de resonancia magnética que mide el nivel de oxígeno en sangre [41]. Nosotros nos enfocamos en las respuestas a los siguientes estímulos: mover la mano; mover el pie; mover la lengua; reconocer formas y caras vistas con anterioridad; clasificar un cuento breve dentro de dos categorías y resolver problemas aritméticos simples. Cada uno de estas tareas se encuentra detallada en el trabajo de Bach et al. [3]. Por cada uno de estos estímulos se cuenta con un mapa de *z-scores* sobre la corteza. Estos *z-scores* representan la correlación entre la activación de cada punto de la corteza con el estímulo.

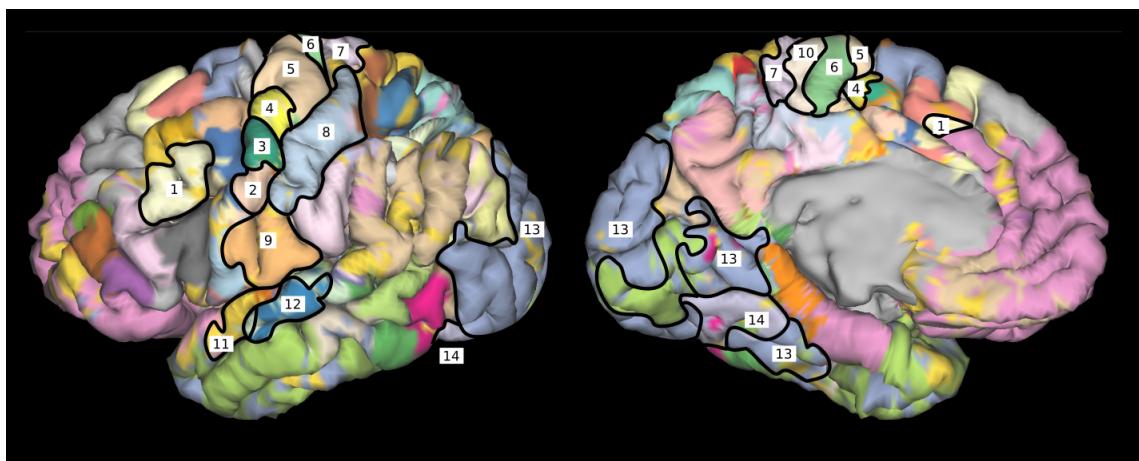


Fig. 6.3: Parcelación obtenida utilizando nuestro método. Se etiquetaron algunas áreas de interés para comparar con un estudio funcional.

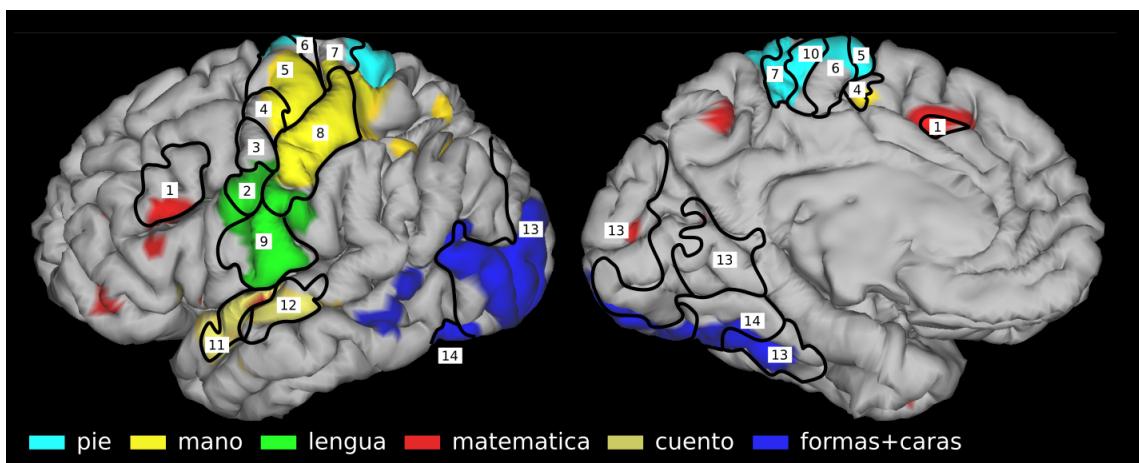


Fig. 6.4: Proyección de 14 áreas obtenidas con nuestro método sobre los *z-scores* mayores a 5 de las activaciones funcionales.

Queremos comparar los mapas de *z-scores* con una de las parcelaciones obtenidas mediante nuestro método. Para ello primero interpolamos la parcelación utilizada en la

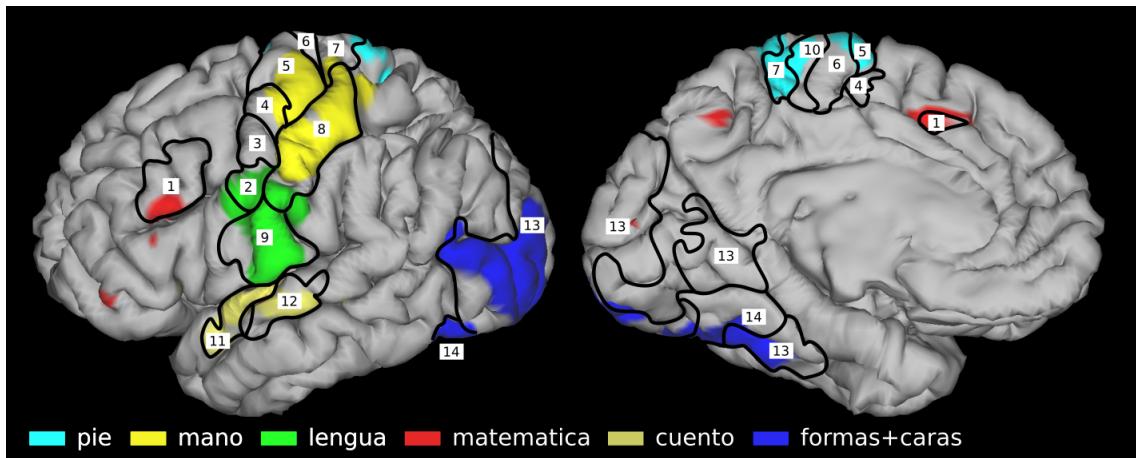


Fig. 6.5: Proyección de 14 áreas obtenidas con nuestro método sobre los *z-scores* mayores a 7 de las activaciones funcionales.

sección anterior a la superficie en que se encuentran los mapas. El resultado de dicha interpolación puede verse en la figura 6.3. Algunas regiones de interés fueron etiquetadas. La figura 6.4 muestra la proyección de estas regiones sobre los *z-scores*  $\geq 5$  de los distintos mapas. La figura 6.5 muestra la proyección sobre los *z-scores*  $\geq 7$ . Luego definimos la siguiente métrica entre superficies:

$$\text{superposicion}(A, B) = \frac{2 * \text{area}(A \cap B)}{\text{area}(A) + \text{area}(B)}$$

Dadas dos superficies  $A$  y  $B$  se cumple:  $0 \leq \text{superposicion}(A, B) \leq 1$ . El 0 representa que son superficies disjuntas y el 1 que están totalmente superpuestas. La tabla 6.1 muestra la superposición entre las superficies de nuestra parcelación y los *z-scores*  $\geq 5$ . La tabla 6.2 muestra la superposición entre las superficies de nuestra parcelación y los *z-scores*  $\geq 7$ . En cada columna se resaltó el valor mas grande. Esto es, el valor de la parcela que mayor superficie compartía con cada mapa de activación.

En el ítem 6.3.1 comparamos la parcelación obtenida por nuestro método contra un atlas anatómico. Ahora lo comparamos con un estudio funcional del sujeto. Veamos una por una las secciones etiquetadas en la figura 6.3 para analizarlas. La región etiquetada como 1 es funcionalmente consistente con la resolución de problemas matemáticos. Las regiones 2 – 10 son consistentes anatómicamente con la corteza motora. A su vez, también son consistentes con el homúnculo cortical, definido por Penfield [40], es una división funcional de la corteza motora y de la corteza somestésica. Mapea distintas regiones del cuerpo con regiones de la corteza como puede verse en la figura 6.6. Las regiones 11 y 12 son consistentes anatómica y funcionalmente con el lóbulo temporal superior. Finalmente, las regiones 13 y 14 son consistentes anatómica y funcionalmente con el lóbulo occipital. Estos resultados son prometedores y motivan a seguir estudiando el método propuesto.

	pie	mano	lengua	matemática	cuento	formas
1	0	0	0	0.276	0	0
2	0	0	0.311	0	0	0
3	0	0	0	0	0	0
4	0	0.172	0	0	0	0
5	0.177	0.321	0	0	0	0
6	0.201	0	0	0	0	0
7	0.231	0.044	0	0	0	0
8	0	<b>0.590</b>	0.139	0	0	0
9	0	0	<b>0.610</b>	0	0	0
10	<b>0.330</b>	0	0	0	0	0
11	0	0	0	0	<b>0.606</b>	0
12	0	0	0	0	0.454	0
13	0	0	0	0	0	<b>0.491</b>
14	0	0	0	0	0	0.154

Tab. 6.1: Relación entre las áreas de cada parcela y los mapas de activación con  $zscore > 5$ . Los valores máximos de cada columna fueron resaltados.

	pie	mano	lengua	matemática	cuento	formas
1	0	0	0	<b>0.419</b>	0	0
2	0	0	0.323	0	0	0
3	0	0	0	0	0	0
4	0	0.115	0	0	0	0
5	0.141	0.341	0	0	0	0
6	0.095	0	0	0	0	0
7	<b>0.377</b>	0.040	0	0	0	0
8	0	<b>0.669</b>	0.147	0	0	0
9	0	0	<b>0.629</b>	0	0	0
10	0.240	0	0	0	0	0
11	0	0	0	0	<b>0.715</b>	0
12	0	0	0	0	0.458	0
13	0	0	0	0	0	<b>0.428</b>
14	0	0	0	0	0	0.154

Tab. 6.2: Relación entre las áreas de cada parcela y los mapas de activación con  $zscore > 7$ . Los valores máximos de cada columna fueron resaltados.

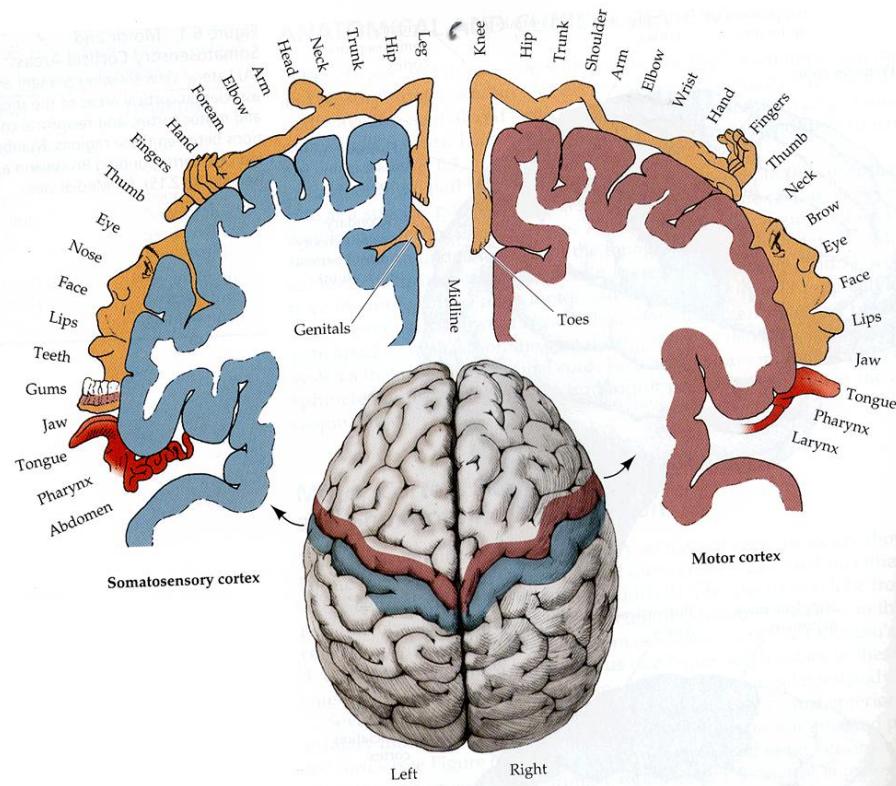


Fig. 6.6: El Homúnculo Cortical es una división funcional de la corteza motora y la corteza somestésica

#### 6.4. Trabajo Futuro

En este trabajo presentamos un nuevo método para parcelar la corteza cerebral mediante el agrupamiento de tractogramas. El método mostró resultados con una fuerte correlación anatómica y funcional sobre el sujeto estudiado. Por eso es importante seguir estudiando algunos aspectos del proceso de parcelamiento. Primero, los tractogramas aquí utilizados fueron generados con el algoritmo 3 de la sección 5.3. Sería interesante estudiar los resultados que se obtienen al usar otros algoritmos de tractografía. Segundo, en las secciones 5.3.1 y 4.4.2 se puede ver que ambos métodos de *clustering* cuentan con un parámetro  $k$ . Este parámetro indica la cantidad de iteraciones a realizar uniendo solo *clusters* vecinos y de tamaño similar. En el trabajo de Moreno-Dominguez [15] muestran una forma de calcular dicho parámetro para su método. Hay que estudiar una forma de calcularlo para el nuestro. Tercero, habría que aplicar el método sobre una población de sujetos y ver su estabilidad. Esto es, estudiar si los resultados son igual de buenos que con el sujeto aquí elegido. Finalmente, en la sección mostramos que es posible aplicar operaciones lineales en el espacio *LogOdds*. Esto puede utilizarse en particular para calcular el tractograma medio de una población de sujetos. En estudios futuros analizaremos el crear un atlas de la corteza basado solo en información estructural.



## Bibliografía

- [1] K. Brodmann, “Vergleichende Lokalisationslehre der Großhirnrinde in ihren Prinzipien dargestellt aufGrund des Zellaufbaues,” 1909.
- [2] M. D. Greicius, B. Krasnow, A. L. Reiss, and V. Menon, “Functional connectivity in the resting brain: A network analysis of the default mode hypothesis,” *Proceedings of the National Academy of Sciences*, vol. 100, pp. 253–258, Jan. 2003.
- [3] D. M. Barch, G. C. Burgess, M. P. Harms, S. E. Petersen, B. L. Schlaggar, M. Corbetta, M. F. Glasser, S. Curtiss, S. Dixit, C. Feldt, D. Nolan, E. Bryant, T. Hartley, O. Footer, J. M. Bjork, R. Poldrack, S. Smith, H. Johansen-Berg, A. Z. Snyder, and D. C. Van Essen, “Function in the human connectome: task-fMRI and individual differences in behavior.,” *NeuroImage*, vol. 80, pp. 169–89, Oct. 2013.
- [4] S. M. Stufflebeam, H. Liu, J. Sepulcre, N. Tanaka, R. L. Buckner, and J. R. Madsen, “Localization of focal epileptic discharges using functional connectivity magnetic resonance imaging,” *Journal of Neurosurgery*, vol. 114, pp. 1693–1697, June 2011.
- [5] K. Oishi, A. V. Faria, P. C M van Zijl, and S. Mori, *MRI Atlas of Human White Matter*. Academic Press, Inc., 2010.
- [6] M. T. de Schotten, “Direct Evidence for a Parietal-Frontal Pathway Subserving Spatial Awareness in Humans,” *Science*, vol. 309, pp. 2226–2228, Sept. 2005.
- [7] J. Song, B. M. Young, Z. Nigogosyan, L. M. Walton, V. a. Nair, S. W. Grogan, M. E. Tyler, D. Farrar-Edwards, K. E. Caldera, J. a. Sattin, J. C. Williams, and V. Prabhakaran, “Characterizing relationships of DTI, fMRI, and motor recovery in stroke rehabilitation utilizing brain-computer interface technology,” *Frontiers in Neuroengineering*, vol. 7, p. 31, July 2014.
- [8] P. Dale, G. J. Augustine, D. Fitzpatrick, W. C. Hall, A.-S. LaMantia, J. O. McNamara, and L. E. White, *Neuroscience. 4th ed.* Sinauer Associates, 2008.
- [9] D. G. Taylor and M. C. Bushell, “The spatial mapping of translational diffusion coefficients by the NMR imaging technique,” *Physics in Medicine and Biology*, vol. 30, pp. 345–349, Apr. 1985.
- [10] P. Hagmann, L. Jonasson, P. Maeder, J.-p. Thiran, V. J. Wedeen, and R. Meuli, “Understanding Diffusion MR Imaging Techniques: From Scalar Diffusion-weighted Imaging to Diffusion Tensor Imaging and Beyond1,” *RadioGraphics*, vol. 26, pp. S205–S223, Oct. 2006.
- [11] M. Descoteaux, R. Deriche, T. Knosche, and A. Anwander, “Deterministic and Probabilistic Tractography Based on Complex Fibre Orientation Distributions,” *IEEE Transactions on Medical Imaging*, vol. 28, pp. 269–286, Feb. 2009.
- [12] S. Jbabdi, M. Woolrich, J. Andersson, and T. Behrens, “A Bayesian framework for global tractography,” *NeuroImage*, vol. 37, pp. 116–129, Aug. 2007.

- [13] J. a. Hartigan and M. a. Wong, "A K-Means Clustering Algorithm," *Journal of the Royal Statistical Society*, vol. 28, no. 1, pp. 100–108, 1979.
- [14] T. Hastie, R. Tibshirani, and J. Friedman, "Springer Series in Statistics The Elements of," *The Mathematical Intelligencer*, vol. 27, no. 2, pp. 83–85, 2009.
- [15] D. Moreno-Dominguez, A. Anwander, and T. R. Knösche, "A hierarchical method for whole-brain connectivity-based parcellation," *Human Brain Mapping*, vol. 35, pp. 5000–5025, Oct. 2014.
- [16] A. Anwander, M. Tittgemeyer, D. Von Cramon, A. Friederici, and T. Knosche, "Connectivity-Based Parcellation of Broca's Area," *Cerebral Cortex*, vol. 17, pp. 816–825, May 2006.
- [17] T. E. J. Behrens, H. Johansen-Berg, M. W. Woolrich, S. M. Smith, C. A. M. Wheeler-Kingshott, P. A. Boulby, G. J. Barker, E. L. Sillery, K. Sheehan, O. Ciccarelli, A. J. Thompson, J. M. Brady, and P. M. Matthews, "Non-invasive mapping of connections between human thalamus and cortex using diffusion imaging," *Nature Neuroscience*, vol. 6, pp. 750–757, July 2003.
- [18] D. C. Van Essen, K. Ugurbil, E. Auerbach, D. Barch, T. E. J. Behrens, R. Bucholz, a. Chang, L. Chen, M. Corbetta, S. W. Curtiss, S. Della Penna, D. Feinberg, M. F. Glasser, N. Harel, a. C. Heath, L. Larson-Prior, D. Marcus, G. Michalareas, S. Moeller, R. Oostenveld, S. E. Petersen, F. Prior, B. L. Schlaggar, S. M. Smith, a. Z. Snyder, J. Xu, and E. Yacoub, "The Human Connectome Project: A data acquisition perspective," *NeuroImage*, vol. 62, no. 4, pp. 2222–2231, 2012.
- [19] P. J. Basser and E. Özarslan, "Introduction to Diffusion MR," in *Diffusion MRI* (H. Johansen-Berg and T. E. J. Behrens, eds.), ch. 1, pp. 2–10, Academic Press, Inc., 2009.
- [20] M. L. Lipton, "Introducing MRI," 2014.
- [21] F. Bloch, "Nuclear Induction," *Physical Review*, vol. 70, pp. 460–474, Oct. 1946.
- [22] H. C. Torrey, "Bloch Equations with Diffusion Terms," *Physical Review*, vol. 104, pp. 563–565, Nov. 1956.
- [23] E. O. Stejskal and J. E. Tanner, "Spin Diffusion Measurements: Spin Echoes in the Presence of a Time-Dependent Field Gradient," *The Journal of Chemical Physics*, vol. 42, no. 1, p. 288, 1965.
- [24] D. Le Bihan and E. Breton, "Imagerie de diffusion in vivo par résonance magnétique nucléaire," *Comptes rendus de l'Académie des sciences. Série 2, Mécanique, Physique, Chimie, Sciences de l'univers, Sciences de la Terre*, vol. 301, no. 15, pp. 1109–1112, 1985.
- [25] P. J. Basser, J. Mattiello, and D. Lebihan, "Estimation of the Effective Self-Diffusion Tensor from the NMR Spin Echo," *Journal of Magnetic Resonance, Series B*, vol. 103, pp. 247–254, Mar. 1994.

- [26] P. T. Callaghan, A. Coy, D. MacGowan, K. J. Packer, and F. O. Zelaya, “Diffraction-like effects in NMR diffusion studies of fluids in porous solids,” *Nature*, vol. 351, pp. 467–469, June 1991.
- [27] D. S. Tuch, “Q-ball imaging,” *Magnetic Resonance in Medicine*, vol. 52, pp. 1358–1372, Dec. 2004.
- [28] J.-D. Tournier, F. Calamante, D. G. Gadian, and A. Connelly, “Direct estimation of the fiber orientation density function from diffusion-weighted MRI data using spherical deconvolution,” *NeuroImage*, vol. 23, pp. 1176–1185, Nov. 2004.
- [29] M. F. Glasser, S. N. Sotropoulos, J. A. Wilson, T. S. Coalson, B. Fischl, J. L. Andersson, J. Xu, S. Jbabdi, M. Webster, J. R. Polimeni, D. C. Van Essen, and M. Jenkinson, “The minimal preprocessing pipelines for the Human Connectome Project,” *NeuroImage*, vol. 80, pp. 105–124, 2013.
- [30] S. Mori and P. C. M. van Zijl, “Fiber tracking: principles and strategies - a technical review,” *NMR in Biomedicine*, vol. 15, pp. 468–480, Nov. 2002.
- [31] C. Thomas, F. Q. Ye, M. O. Irfanoglu, P. Modi, K. S. Saleem, D. a. Leopold, and C. Pierpaoli, “Anatomical accuracy of brain connections derived from diffusion MRI tractography is inherently limited,” *Proceedings of the National Academy of Sciences*, vol. 111, pp. 16574–16579, Nov. 2014.
- [32] J. a. Sethian and A. Vladimirs, “Ordered upwind methods for static Hamilton-Jacobi equations,” *Proceedings of the National Academy of Sciences*, vol. 98, pp. 11069–11074, Sept. 2001.
- [33] F. Murtagh, *Multidimensional Clustering Algorithms*. Vienna: Comps Physica Verlag, 1985.
- [34] B. Efron, *The Jackknife, the Bootstrap and Other Resampling Plans*. Society for Industrial and Applied Mathematics, Jan. 1982.
- [35] I. Aganj, C. Lenglet, G. Sapiro, E. Yacoub, K. Ugurbil, and N. Harel, “Reconstruction of the orientation distribution function in single- and multiple-shell q-ball imaging within constant solid angle,” *Magnetic Resonance in Medicine*, vol. 64, pp. n/a–n/a, June 2010.
- [36] M. Descoteaux, E. Angelino, S. Fitzgibbons, and R. Deriche, “Regularized, fast, and robust analytical Q-ball imaging,” *Magnetic Resonance in Medicine*, vol. 58, pp. 497–510, Sept. 2007.
- [37] K. M. Pohl, J. Fisher, S. Bouix, M. Shenton, R. W. McCarley, W. E. L. Grimson, R. Kikinis, and W. M. Wells, “Using the logarithm of odds to define a vector space on probabilistic atlases,” *Medical Image Analysis*, vol. 11, pp. 465–477, Oct. 2007.
- [38] R. E. Bank and C. C. Douglas, “Sparse matrix multiplication package (SMMP),” *Advances in Computational Mathematics*, vol. 1, pp. 127–137, Feb. 1993.

- [39] R. S. Desikan, F. Ségonne, B. Fischl, B. T. Quinn, B. C. Dickerson, D. Blacker, R. L. Buckner, A. M. Dale, R. P. Maguire, B. T. Hyman, M. S. Albert, and R. J. Killiany, “An automated labeling system for subdividing the human cerebral cortex on MRI scans into gyral based regions of interest,” *NeuroImage*, vol. 31, pp. 968–980, July 2006.
- [40] W. Penfield and H. Jasper, *Epilepsy and the Functional Anatomy of the Human Brain*. 1954.
- [41] S. Ogawa, T. M. Lee, A. R. Kay, and D. W. Tank, “Brain magnetic resonance imaging with contrast dependent on blood oxygenation.,” *Proceedings of the National Academy of Sciences*, vol. 87, pp. 9868–9872, Dec. 1990.